Doctoral theses at NTNU, 2019:185

Francesco Valerio Gianni

From Ideation to Prototyping of IoT Systems: The Case of Smart Cities

Doctoral thesis

NTNU
Norwegian University of Science and Technology
Thesis for the Degree of Philosophiae Doctor
Faculty of Information Technology and Electrical Engineering
Department of Computer Science

NTNU
Norwegian University of Science and Technology

Francesco Valerio Gianni

Doctoral theses at NTNU, 2019:185

ISBN 978-82-326-3966-3 (printed ver.)
ISBN 978-82-326-3967-0 (electronic ver.)
ISSN 1503-8181

NTNU
Norwegian University of Science and Technology

NTNU

Francesco Valerio Gianni

From Ideation to Prototyping of IoT Systems: The Case of Smart Cities

Thesis for the Degree of Philosophiae Doctor

Trondheim, June 2019

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Computer Science

NTNU
Norwegian University of
Science and Technology

NTNU
Norwegian University of Science and Technology

Thesis for the Degree of Philosophiae Doctor

Faculty of Information Technology and Electrical Engineering
Department of Computer Science

ISBN 978-82-326-3966-3 (printed ver.)
ISBN 978-82-326-3967-0 (electronic ver.)
ISSN 1503-8181

Doctoral theses at NTNU, 2019:185

Printed by NTNU Grafisk senter

*'You Improvise,
You Adapt,
You Overcome'*

Clint Eastwood — Heartbreak Ridge, 1986

Abstract

Interacting with computers, for work or for leisure, originally required humans to *speak the language of the machine*, intended as interaction techniques, languages and procedures that are closer to how a computer works rather than how the human brain functions. Nowadays, despite the wide availability of computers, this gap is still present. It is now easier to use computers; however, programming even simple applications is still a task approachable only through complex jargon, which has little in common with the final scope of the application. By addressing the Internet of Things (IoT), we found that this situation is complicated by the involvement of electronics, microcontrollers and low-level programming languages. Specific professional skills are often required to develop and prototype IoT applications. Research on human-computer interaction (HCI) addresses the challenges of interconnecting people and computers, building tools and theories to facilitate the many uses that a computer can have, in an open-ended dialogue. Thanks to research in this field, new solutions and interaction strategies allowed us to improve user experience when dealing with computers. However, owing to the complexity of the field, in IoT, it is still challenging to keep humans in the loop, in terms of both the development of IoT applications and their use. Specific branches in HCI aim to facilitate the programming phase of computer applications; however, despite being simple, such process still requires the user to have some non-trivial technical skills and an understanding of the basic logical constructs common to most programming languages.

In this thesis, I address how to empower new audiences in brainstorming, designing, programming and prototyping applications for IoT. Already established research fields, such as end-user development (EUD), HCI, interaction design (ID) and software engineering, have already investigated some of these challenges. In this thesis, I will restrict the domain to a specific subset of IoT applications based on tangible user interfaces (TUIs) and smart objects, covering the phases of brainstorming and design of such applications. Particular focus will be placed on promoting smart city learning (SCL) through the IoT applications envisioned. SCL explores how citizens can be actively involved in a learning process that occurs in the city, making use of its data and services, in order to increase awareness and lifelong learning. The SCL application domain was chosen since it can benefit from IoT technologies, promoting inclusion and participation. In fact, IoT is scarcely used in applications for SCL, while citizens struggle to contribute actively in the life of the city. In order to achieve this inclusive vision, the entry barriers to technology need to be lowered and an integrated process guiding the users during all the phases of the development process should be devised. In this thesis, I will explore how concepts from design thinking, HCI and user-centred design can be implemented in novel tools and processes to support such goal.

This work is grounded in design science research methodologies. Several prototypes were built during eight design iterations, and field studies have been performed at the end of each iteration, for evaluation and validation against acceptance, usability and impact on the problem at stake. Software and hardware rapid prototyping techniques and open-source and digital manufacturing tools have been largely employed. The results of the evaluations were used for validating the already existing theories in HCI, SCL and IoT and for the development of new constructs. Commercial exploitation of the research outcomes is underway.

The resulting contributions add new knowledge to guide the ideation, design, programming and prototyping of IoT applications for users with limited technical skills, placing particular emphasis on the case of applications for SCL. A holistic framework for IoT has been devised to bootstrap the ideation process and guide the users towards a tangible, programmable prototype of the application idea. The work described in this thesis includes the implementation of two toolkits for ideation, programming and physical prototyping. This development task required a wide range of competencies in design, software and hardware engineering. Some of these critical competencies were acquired by the author during the course of the investigation. The lessons learned from the experience of the author provided knowledge connected to the creation of tools to ease idea generation and rapid prototyping of IoT applications.

Preface

This thesis was submitted to the Norwegian University of Science and Technology (NTNU) for partial fulfilment of the requirements for the degree of philosophiae doctor.

The main body of doctoral work has been performed at the Department of Computer Science, NTNU, Trondheim, Norway. Professor Monica Divitini was the main supervisor, and Professor John Krogstie and Dr Simone Mora were the co-advisers.

Acknowledgements

Most of all, I would like to express my deepest gratitude to my supervisor Professor Monica Divitini. You have always been present and supportive for me. Under your guidance I grew up as a researcher and as a person. Thanks for the patience and the effort you dedicated to me.

I am extremely grateful to my co-advisors Professor John Krogstie and Dr Simone Mora. You have been an example of integrity and professionalism, I was inspired by your hard work and always found support and advice when in need.

I wish to thank all the colleagues and co-authors in our research group who contributed to my work: Anna, Lisa, Mikael and Dag Rune.

I extend my gratitude to the colleagues from the Department of Computer Science and the ISSE group, especially to my mates Elena, Stefano, Massimiliano, Katerina and Sofia for the engaging and constructive discussions.

Thanks to all my friends in Norway and in Italy for always being supportive, curious and creative, you inspired me to strive for more.

Thanks to my family: Valeria, Massimo, Matteo, Daniele, and to my grandparents Piero, Piera, Cesi and Ugo. Without you all of this would not have been possible, each of you built a piece of who I am today.

A big thank you to Cesilie, who had the rare privilege of being the one to stand *grumpy*-me, *late-working*-me and *ranting*-me. Thanks for patiently listening, for always having helpful words and for the hugs of encouragement.

Francesco Gianni
26th May 2019

Contents

Quotes iii

Abstract v

Preface vii

Acknowledgements ix

Contents xi

List of Figures xv

List of Tables xvii

Abbreviations xix

I Part I 1

1 Introduction 1
1.1 Domain . . . 1
1.1.1 Internet of Things . . . 1
1.1.2 Non-Expert Users . . . 2
1.2 Challenges . . . 2
1.3 Motivation . . . 3
1.4 Problem Statement . . . 4
1.5 Research Methodology . . . 6
1.6 Research Questions . . . 6
1.7 Research Outcomes . . . 7
1.7.1 Research Papers . . . 7
1.7.2 Research Contributions . . . 8
1.8 Context of the Work . . . 9
1.9 Structure of the Thesis . . . 10

2 The Case of Smart City Learning 11
2.1 Smart Cities . . . 11

2.2 Citizen Participation 12
2.3 Learning 13
2.4 Technology 13
2.5 SCL as a Domain Case 14

3 Related Work 15
3.1 Common Toolkit Architectures 16
3.2 Card-Based Toolkits for IoT 16
3.3 End-User Development for IoT 18

4 The Tiles IoT Framework 19
4.1 Process 19
4.2 Tools 20
4.2.1 The Tiles Ideation Toolkit 22
4.2.2 The RapIoT Toolkit 23
4.3 Integration of the Toolkits in Prototyping 27
4.4 Lowering the Bar: Non-Experts as Target Users 28
4.5 Identified Values of the Toolkits 29
4.6 Learning from the Past: Tackling Current IoT Challenges 30
4.7 Vision for the Tiles IoT Framework 31

5 Theoretical Underpinning 33
5.1 Co-Design 33
5.2 Tangible User Interaction with Smart Objects 34

6 Research Methodology 37
6.1 Research Philosophy 37
6.2 Design as Research 37
6.2.1 Relevance Cycle 38
6.2.2 Rigor Cycle 39
6.2.3 Design Cycle 39
6.3 Research Strategy and Methodological Choice 39
6.4 Research Activities 39
6.4.1 User Studies 41
6.4.2 Design Iterations 42
6.5 Research Approach in Field Studies 45
6.5.1 Construct Validity 45
6.5.2 Internal Validity 46
6.5.3 External Validity 47
6.5.4 Reliability 47
6.6 Ethical Considerations 48
6.7 User Centred Design Approach 49

7 Results 51
7.1 Overview of the Research Papers 51

7.2 P1: Technology-Enhanced Smart City Learning: A Systematic Mapping of the Literature. . . . 51
7.3 P2: Tiles: A Card-Based Ideation Toolkit for the Internet of Things. 52
7.4 P3: Designing IoT Applications for Smart Cities: Extending the Tiles Ideation Toolkit. . . . 53
7.5 P4: Tiles-Reflection: Designing for Reflective Learning and Change Behaviour in the Smart City. . . . 54
7.6 P5: Designing IoT Applications in Lower Secondary Schools. 55
7.7 P6: RapIoT Toolkit: Rapid Prototyping of Collaborative Internet of Things Applications. . . . 56
7.8 P7: Rapid Prototyping Internet of Things Applications for Augmented Objects: The Tiles Toolkit Approach. . . . 58

8 Contributions 61
8.1 C1: Improved understanding of the holistic process supporting non-experts in idea generation, design and prototyping of IoT applications 63
8.2 C2: Contribution to the design, implementation and evaluation of a card-based design toolkit for idea generation, allowing non-experts to generate concepts of IoT applications 64
8.3 C3: Contribution to the design, implementation and evaluation of a rapid prototyping toolkit for tangible exploration of ideas based on smart augmented objects 65
8.4 C4: Improved understanding of educational methods to support learning about smart cities and IoT 66
8.5 C5: Knowledge in the definition of a technological framework to overcome recognised challenges and limitations affecting the IoT domain 66

9 Evaluation 67
9.1 Evaluation of Research Questions 67
9.2 Limitations of the Research Approach 68
9.2.1 Case Studies 68
9.2.2 Role of Learning 69
9.3 Exploitation and External Impact of Research 70

10 Conclusions and Future Work 71

References 75

II Part II 83

List of Figures

4.1 The process envisioned by the Tiles IoT framework. 19
4.2 Mapping of toolkits, components and phases of the Tiles IoT framework. 21
4.3 The last version of the Tiles ideation toolkit. The cardboard and the cards, including the extensions described in P3, are visible in the picture. 23
4.4 The cardboard included in the Tiles ideation toolkit. The circled sections are connected to the phases of the Tiles IoT framework process. 24
4.5 Cards included in the Tiles ideation toolkit. 25
4.6 Electronic *stickers* employed by the users to prototype the smart objects. 26

5.1 A visual example of how a simple design can be easy to use, whereas a more flexible and feature-rich one is less usable (Lidwell, Holden and Butler, 2010, p. 103). 36

6.1 Schema of the three design science research cycles, based on Hevner's model (2007). 38
6.2 Video recording of the user interaction occurring around the Tiles cards and cardboard. The brainstorming workshop pictured was one of the studies included in P4. 40
6.3 A design workshop with university students, part of the studies included in P3. 41
6.4 Details of the CAD circuit board of the Tiles Temp v2. The temperature and humidity sensors are located in the two dashed areas. 44

8.1 A schema of the contributions. On top, from left to right: the domains of SCL, ID, HCI and IoT. 62

List of Tables

1.1 Connection between research papers and research questions. 8

6.1 List of prototypes built 43

8.1 Mapping of contributions with papers. 63

Abbreviations

NTNU Norwegian University of Science and Technology

ICT Information and Communication Technologies

UCD User Centred Design

API Application Programming Interface

DSL Domain Specific Language

IDE Integrated Development Environment

I/O Input/Output

W3C World Wide Web Consortium

CSRL Computer Supported Reflective Learning

HCI Human-Computer Interaction

ID Interaction Design

IoT Internet of Things

M2M Machine to Machine

GUI Graphical User Interface

TUI Tangible User Interface

IP Internet Protocol

SCL Smart City Learning

EUD End-User Development

Part I

Introduction and Methods

1 Introduction

1.1 Domain

In this section the research domain of the thesis and the target user group are presented, and a brief description will introduce the adopted concepts of the *Internet of Things (IoT)*, *smart cities* and *non-expert* users.

1.1.1 Internet of Things

Connected objects and appliances appeared as early as in 1982, with the first connected coke machine developed at Carnegie Mellon University. Since then, many definitions have been proposed for IoT (Ashton, 2009; Gubbi et al., 2013), while a few basic principles still stand at the foundation and are shared among most of the theoretical interpretations:

- Computers are no longer objects, but accessories that can be embedded or placed in the environment, deploying computational power away from the desktop or the server room.
- IoT devices have a connected nature. They can share and receive data from each other or communicate over the internet.
- Computers are now capable of sensing and acting in the physical world, which goes beyond the traditional computer paradigm of mice and keyboards as input devices and screens as output devices.

Some of the proposed definitions are connected to the use of particular technological approaches (Welbourne et al., 2009), whereas others are related to the domain or how the technology is used. A few examples involve machine-to-machine (M2M) systems, networks of sensors to collect data on a territory (Murty et al., 2008) and technology-augmented appliances designed for a particular domain like households (Alkar and Buhur, 2005) or cities (Hernández-Muñoz et al., 2011).

In my research, I focused on IoT as an approach to *object augmentation*, as a technological medium to enable the creation of *smart objects*. Given the exploratory nature of the research work performed, energy efficiency, security and cost-reduction aspects related to IoT were addressed with a low level of priority. Forms of IoT outside the notion of smart objects, like the above-mentioned sensor networks, screen-based appliances and IoT applications that do not foresee any user involvement, remained outside of my research domain.

Smart objects appear and can be used like a regular object, but at the same time they provide additional affordances, thanks to their embedded technological layer. They have been previously defined by Michael Beigl and his colleagues as '*everyday artefacts augmented with computing and communication, enabling them to establish and exchange information about themselves with other artefacts and/or computer applications*' (Beigl, Gellersen and Schmidt, 2001). This approach to IoT through smart objects enables the creation of *tangible user interfaces* (TUIs) (Ishii and Ullmer, 1997), a method to interact with a computer application using object manipulation instead of a keyboard and to use output feedback involving several senses instead of a screen.

1.1.2 Non-Expert Users

During my research, the technological approach I embraced was based on a user-centred perspective, thus promoting inclusion and participation, extending the impact of the solutions to a wide portion of the population. I addressed a precise target group of users, defined as *non-experts*, who are individuals that do not need to possess professional skills in design, programming languages, electronics or computer networks.

The aim of my work was to directly involve non-experts in designing meaningful IoT applications, through a process tailored to their capabilities, while meeting their needs and desires. The ultimate goal was to provide access to the affordances offered by IoT to new communities, lowering the access barriers and reducing the complexity.

1.2 Challenges

Working with TUIs and smart objects is challenging because building meaningful applications requires skills in multiple domains, like human-computer interaction (HCI), design, programming and electronics. Moreover, works in these fields have traditionally focused on technical facets (Siegemund, 2004), whereas HCI aspects received attention only recently (Nelson and Churchill, 2005). Yet, design principles and methods for smart objects that go beyond mere hardware are at the forefront of research exploration (Kortuem et al., 2010).

Another demanding point is related to the development process: heterogeneous needs from end-users make the development of products and technological solutions increasingly difficult (Von Hippel, 2001). Toolkits for innovation address these challenges, allowing end-users to play an active role in product development. Toolkits allow breaking down the design space into atomic tasks and building blocks, which can be more easily recombined allowing *design-by-trial-and-error*, avoiding costly iterations and speeding up the process (Cvijikj and Michahelles, 2011).

Despite the simplifications introduced by the use of the toolkits, following a user-centred approach can still be hard. Non-experts belong to a diverse set of categories, characterised by different needs and skills. Collaborative processes can help by stimulating discussions and allowing users to reach a common ground, a shared lingo allowing them to contribute to the design process despite their diversity.

1.3 Motivation

Many solutions and toolkits have been proposed to facilitate the development of IoT applications (Udoh and Kotonya, 2017). The technologies supporting this process include software, hardware, electronic devices and networking protocols. Efforts were made to simplify the assembly and programming of electronic devices involving the use of sensors, actuators and network connectivity. On the software side, several organisations have proposed different standards to define a common interface and a shared data model to represent the typical information bits related to a wide number of IoT devices. Numerous networking protocols were also created, adapted or re-used in the IoT domain to address specific trade-offs involving bandwidth, latency, energy consumption and wireless coverage (Chen and Kunz, 2016). In terms of hardware, Arduino (Mellis et al., 2007) started a revolution introducing a platform that allows users with limited electronic skills to wire sensors and actuators to a microcontroller, programmed through a simplified software toolchain compared to approaches traditionally used in embedded programming. Despite the innovative solutions and significant advancements in these sub-fields, important challenges are still present in each of them, especially when targeting new audiences that might lack the domain knowledge and technical skills traditionally required to start working with the IoT.

For example, despite the numerous standards, data models, interfaces and networking protocols proposed, none of them emerged as a clear winner, providing an open, royalty-free solution on a par with HTTP and HTML for the web (Guinard and Trifa, 2016). In addition, the data models and software libraries proposed only address IoT devices designed to sense and interact with the ambient environment, such as a thermostat, a temperature sensor or a lamp. There is little support for keeping humans in the loop through typical gestures and patterns that characterise human interaction with objects, like touching, shaking, tilting and so on. On the hardware side, basic electronic skills and knowledge are still required to wire electronic components to an Arduino microcontroller.

Already existing IoT toolkits and frameworks usually focus on facilitating a single aspect of the development process of an IoT application. For example, they assume that the user will continue in such a process by choosing and adopting a software framework to program the application logic for the electronic device assembled, or that the user will solve the network connectivity problem independently. This silo effect is particularly challenging for non-experts, who might lack the critical skills required to address a particular aspect.

With IoT being a loosely defined concept, non-experts often need to start learning about the domain and technology, exploring the solution space while brainstorming a possible application idea. The silo effect can be mitigated by gradually building knowledge, utilising the same abstractions and constructs during all the development phases. This approach helps devise a holistic process rather than an arbitrary selection of tools that are not necessarily designed to work together. A holistic approach can then ease the democratisation of the technology, enabling wider audiences.

The main domain case addressed in this thesis undertakes some of the challenges affecting modern smart cities. In fact, citizens often fall into the definition of non-experts. Few studies have explored how to enable co-design and promote citizen participation through the use of IoT technologies that keep humans in the loop. A high level of participation in acknowledging the problem and developing a technological solution also promotes the appropriation of the idea. The benefits extend to the idea adoption and to the achievement of a learning impact that lasts longer, compared to prescriptive methods.

1.4 Problem Statement

The goal of my work was to find new ways to empower non-experts in the development of IoT applications. This was framed as a holistic and creative process, which also aimed to promote lifelong learning and awareness about opportunities and challenges affecting modern smart cities. Smart city learning (SCL) was used as a case and as an application domain. It helped the users find design constraints and goals while at the same time providing a playground to tackle widely recognised societal challenges affecting citizens worldwide. For my research, SCL has demonstrated to be a compelling domain case since (i) citizens often belong to the non-expert user category, (ii) IoT and smart objects can help support lifelong learning, (iii) SCL keeps a user-centred perspective and (iv) it addresses challenges affecting a wide portion of the population.

Analysing the literature on the technological applications for SCL, we discovered a lack of research regarding the following:

- The use of IoT, tangible interfaces and smart objects, defined in Section 1.1. Most applications make use of smartphones or screen-based interfaces or do not require direct user interaction with the technology.
- The role of smart cities as an evolving community of citizens that cooperate and learn continuously to solve challenges related to large urbanisation. Research on smart cities is often limited to confined sub-domains or restricted communities.

- Active engagement of the users and co-design in the city. Users are often only marginally involved in the studies, resulting in a little impact in terms of learning, long-term behaviour change or awareness of the city challenges.

- When addressing behaviour changes, the methodologies employed could not convey knowledge or awareness or enrich the user in any way. Most of the times, prescriptive methodologies such as persuasion were used. These methods have limited potential in stimulating the learning outcome and, as such, achieving results that last in time.

The tools, theories and technological artefacts adopted during my PhD explored the opportunities at the intersection of the above-mentioned fields. The solutions envisioned follow the principles detailed here:

- **IoT** - Technological applications make use of augmented objects and tangible interfaces. Those applications orchestrate the use of ecologies of different interconnected objects. Such objects support the user experience and the application objective by providing immediate sensory feedback, detecting user interaction and collecting environmental data, while being able to communicate with Internet services and data providers.

- **SCL** - The domain of smart cities is seen from a user-centred perspective. Several challenges affecting the world population living in cities have been identified by the United Nations (UN, 2015). These challenges were proposed as domain problems to non-expert users, providing a design goal for the IoT applications to be created.

- **Co-Design** - The envisioned process extends the involvement of the users beyond the simple usage of the application, addressing also the development phase. Given the potential complexity of the IoT and the non-expert nature of the target users, supporting co-design in this context is particularly challenging.

- **Behaviour Changes and Lifelong Learning** - The IoT applications created by the users during co-design activities are designed to support lifelong learning. Behaviour changes are encouraged and focus on reflective learning and increased awareness. Learning also includes knowledge about the domain and the technologies, which is conveyed while brainstorming and designing the IoT application. These methods better promote a long-term impact and a more permanent outcome, in relation to the specific problems or challenges addressed.

1.5 Research Methodology

The research methodology adopted is based on *design science research* (Hevner and Chatterjee, 2010; March and Smith, 1995). Exploratory studies were conducted mainly in the form of design and prototyping workshops, following a *user-centred approach* (Maguire, 2001; Gulliksen et al., 2003). During these activities, theories and tools developed during multiple iterations were validated on the field. Co-design was used as a strategy to pursue collaboration, awareness and a long-term impact for the end-users.

Quantitative and qualitative research methods (Robson, 1993) have been adopted, including observations of the activities of the end-users during the studies. Quantitative data were collected in the form of questionnaires and through a systematic analysis of the artefacts produced by the users. During the studies, the users also produced scenarios, personas, storyboards and public pitch of ideas. These user-generated materials aided the user-centred design work of improvement and refinement of the tools and methods employed. Consistent with the design science research methodology, grounded in the activities of *building* artefacts for a specific purpose and of *evaluating* how well the artefacts perform (March and Smith, 1995), a number of prototyping iterations and evaluation studies have been performed on the tools employed during the user studies.

Prototyping involved the construction of a set of tools and technologies supporting the various stages of the development process of an IoT application. The design of such tools was grounded in relevant theories and refined by the experience matured during the user studies, which also contributed to the validation of theories and to the development of new constructs. The numerous user studies performed facilitated building and understanding the domain. I did not have any previous knowledge of design methods and had limited knowledge of electronic prototyping.

All the tools produced were evaluated during studies with the end-users; some of the tools went through multiple iterations. The goal was to assess the utility, usability and efficacy.

1.6 Research Questions

The main research question for my PhD work is as follows:

> **MRQ:** How can non-expert users be supported during brainstorming, design and tangible exploration of ideas of IoT applications for SCL?

In order to answer the main research question, this work was broken down into three sub-questions:

> **RQ1:** How can the phases of brainstorming, design and tangible exploration of IoT ideas be connected and integrated to provide an experience adapted to the skills of non-expert users?

RQ2: Which kind of brainstorming and design tools can be employed to generate IoT application ideas? And how can they be specialised to target specific goals and application domains?

RQ3: Which IoT technologies and architectures can efficiently support prototyping and tangible exploration of ideas for non-expert users?

1.7 Research Outcomes

Seven research papers published in peer-reviewed conferences and journals explored the research questions. Building on the results reported in these papers, a body of knowledge regarding the research questions in the fields of SCL, interaction design (ID), HCI and IoT has been developed.

Finally, actions were taken to publicly release the solutions developed, while research contributions were evaluated for commercial exploitation. More information regarding these matters is provided in Section 9.3.

1.7.1 Research Papers

The research questions are addressed in the following research papers:

P1 Gianni, Francesco, and Monica Divitini (2016) **'Technology-Enhanced Smart City Learning: A Systematic Mapping of the Literature.'** *Interaction Design and Architecture(s) Journal - IxD&A* 27:28–43.

P2 Mora, Simone, Francesco Gianni, and Monica Divitini (2017) **'Tiles: A Card-Based Ideation Toolkit for the Internet of Things.'** In: Proceedings of the 2017 Conference on Designing Interactive Systems. DIS 2017. 587–598. Edinburgh, United Kingdom: ACM.

P3 Gianni, Francesco, and Monica Divitini (2018) **'Designing IoT Applications for Smart Cities: Extending the Tiles Ideation Toolkit.'** *Interaction Design and Architecture(s) Journal - IxD&A* 35:100–116.

P4 Gianni, Francesco, Lisa Klecha, and Monica Divitini (2019) **'Tiles-Reflection: Designing for Reflective Learning and Change Behaviour in the Smart City.'** In: The Interplay of Data, Technology, Place and People for Smart Learning. SLERD 2018. *Smart Innovation, Systems and Technologies* 95:70–82. Aalborg, Denmark: Springer, Cham.

P5 Mavroudi, Anna, Monica Divitini, Francesco Gianni, Simone Mora and Dag R. Kvittem (2018) **'Designing IoT Applications in Lower Secondary Schools.'** In: Proceedings of IEEE Global Engineering Education Conference. EDUCON 2018. 1120–1126. Tenerife, Spain: IEEE.

🏆 – *Best Paper Award at EDUCON 2018.*

P6 Gianni, Francesco, Simone Mora, and Monica Divitini (2018) **'RapIoT Toolkit: Rapid Prototyping of Collaborative Internet of Things Applications.'** *Journal of Future Generation Computer Systems.* Elsevier.

🏆 – *The conference version of this paper received the Best Paper Award at the International Conference on Collaboration Technologies and Systems (CTS) in 2016.*

P7 Gianni, Francesco, Simone Mora, and Monica Divitini (2018) **'Rapid Prototyping Internet of Things Applications for Augmented Objects: The Tiles Toolkit Approach.'** In: Ambient Intelligence. AmI 2018. *Lectures Notes in Computer Science.* 11249:204–220. Larnaca, Cyprus: Springer, Cham.

Table 1.1 shows the mapping between research papers and research questions.

Table 1.1: Connection between research papers and research questions.

	P1	P2	P3	P4	P5	P6	P7
RQ1		•	•			•	•
RQ2	•	•	•	•	•		
RQ3						•	•

1.7.2 Research Contributions

The seven papers published added to the following contributions:

> ***C1:*** *Improved understanding of the holistic process supporting non-experts in idea generation, design and prototyping of IoT applications.* This presents challenges and lessons learned derived from the field experience of the author in designing and evaluating a toolkit to support such process.

> ***C2:*** *Contribution to the design, implementation and evaluation of a card-based design toolkit for idea generation, allowing non-experts to generate concepts of IoT applications.* This includes the evaluation, refinement and specialisation of the toolkit to better target the SCL domain.

C3: *Contribution to the design, implementation and evaluation of a rapid prototyping toolkit for tangible exploration of ideas based on smart augmented objects.* This includes the design and production of wireless electronic devices, applications for mobile devices and cloud-based software.

C4: *Improved understanding of educational methods to support learning about smart cities and IoT.* This investigates workshops for brainstorming and tangible exploration of IoT ideas as a means to convey knowledge about technology and smart cities.

C5: *Knowledge in the definition of a technological framework to overcome recognised challenges and limitations affecting the IoT domain.* This describes the first draft of an IoT framework supporting applications that involve object manipulation, as well as sensory feedback and data collection. How the framework can be implemented using established standards and open technologies is also described.

1.8 Context of the Work

The research work presented in this thesis contributed to the development of the Tiles project[1]. The goal of this project is to design and manufacture an integrated set of tools to enable new, non-technical audiences to brainstorm, design and prototype IoT applications involving smart objects. Tiles propositions include ease of use, support for multiple design iterations, velocity, rapid feedback loops and user enjoyment. As a contributing researcher in the Tiles project, I took part in different activities, such as design, implementation, development coordination, revision and evaluation of different tools created in the context of the project.

The tools produced and the data collected during the evaluations were also employed to explore the learning potential of the Tiles approach in educational contexts. This work was conducted thanks to the support from partners in the UMI-Sci-Ed project[2], funded by the European Union. The outcomes of such research are valuable for the work described in this thesis, since they provided insights into the ability of the tools created to convey knowledge about IoT and smart cities to children. Such domain knowledge was required to inform the subsequent phases of development of the IoT ideas.

During my PhD period, I co-advised the specialisation project and master thesis of two students who contributed to the Tiles project and to other areas. The co-advising activities resulted in the co-authoring of P4 and two other articles. In addition, I coordinated the development of the mobile application described in P6. The work was performed by a group of students, as part of a semester course on the management of real-world technical projects.

[1] www.tilestoolkit.io
[2] www.umi-sci-ed.eu

1.9 Structure of the Thesis

The thesis is composed of two parts:

- **Part I** – presents the introduction to the research work and provides an overview of the background theories, the methods used, the results achieved and the contributions made by the thesis.
- **Part II** – contains the seven research papers that added to the results of this thesis.

The rest of **Part I** is organised as follows:

- **Chapter 2** offers an overview of SCL and motivates its employment as an application domain.
- **Chapter 3** introduces related work and defines the toolkits for IoT.
- **Chapter 4** presents and defines the IoT framework embraced in terms of components, design phases and boundaries.
- **Chapter 5** describes constructs adopted as theoretical underpinning: co-design and the HCI approach based on TUIs.
- **Chapter 6** depicts the research method and approach adopted, providing an overview of the user studies conducted and the prototypes built.
- **Chapter 7** summarises the results of the research papers.
- **Chapter 8** outlines the contributions of the thesis and their relation to the research papers.
- **Chapter 9** proposes an evaluation of the contributions in relation to the research questions.
- **Chapter 10** concludes the thesis sketching future research and possible evolution of the tools presented.

Part II contains the seven research papers in full length.

2 The Case of Smart City Learning

In this chapter, I introduce the domain of *SCL*, used as a case and an application domain. SCL was employed as a strategy to promote sustainability, awareness and lifelong learning about the challenges affecting modern smart cities. It was used as a design goal, proposed by researchers or facilitators, but directly pursued by the users involved in the studies.

In the remaining of the thesis, the technical solutions described in Chapter 4 and theoretical frameworks presented in Chapter 5 complement the SCL vision by describing how practical applications can be deployed in the field.

2.1 Smart Cities

Nowadays, most of the world's population live in cities. While cities occupy only 2% of the surface of the world, they are responsible for 75% of the global energy consumption and 80% of CO_2 emissions. The concept of smart cities arose as a means to foster innovation and technology adoption and develop data networks in urban environments. Smart city applications focus on a wide range of sub-domains like governance and service infrastructure (i.e. smart grids and water management) or on collecting and utilising data. Smart cities are founded on communities of citizens; the social aspect plays an important role in ensuring a meaningful deployment of the technology, which should serve the citizens despite their diversities.

One of the most cited definitions in this regard is the one advanced by Caragliu, Del Bo and Nijkamp (2011), who stated that a city is smart '*when investments in human and social capital and traditional (transport) and modern (ICT) communication infrastructure fuel sustainable economic growth and a high quality of life, with a wise management of natural resources, through participatory governance*'.

ICT has recently become part of the mainstream debate on urban sustainability as well as urbanisation because of the ubiquity presence of urban computing and the massive use of urban ICT in urban systems and domains (Bibri and Krogstie, 2017b). Indeed, data sensing and information processing are being fast embedded into the very fabric of contemporary cities (Batty et al., 2012; Bibri and Krogstie, 2016). A large number of advanced technologies are being developed and applied in response to the urgent need for dealing with the complexity of the knowledge necessary for enhancing, harnessing and integrating urban systems and facilitating collaboration and coordination among urban domains in the realm of smart sustainable urban planning and development (Bibri and Krogstie, 2016).

The emergence of this new techno-urban phenomenon has been particularly fuelled by what is labelled 'ICT of the new wave of computing', that is, a combination of various forms of pervasive computing, the most prevalent of which are ubiquitous computing, ambient intelligence (AmI), IoT and sentient computing (Bibri and Krogstie, 2017a).

In my work, I envision a smart city firstly as a community of citizens that live and move in the urban environment, as part of their daily routine. This vision finds its roots in the work of Hollands (2008), aiming to support regional competitiveness. Technology is seen as an enabling factor, which does not disrupt the activities of the citizens but rather encourages awareness, builds problem-solving skills and supports lifelong learning, towards an improved quality of life.

2.2 Citizen Participation

Despite the volume of research targeting smart cities, citizens, especially young ones, are often included only for symbolic purposes; meaningful inclusion remains an open challenge (Lansdown, 2010). While conventional methods of public participation like committee groups and public hearings have failed to engage the majority of the public (Roberts, 2004; Irvin and Stansbury, 2004), multidisciplinary methods are currently being investigated in order to give voice to citizens and stakeholders through authentic dialogue, building social capital and trust (Innes and Booher, 2004).

Previous research has voiced the need for ICT development and innovation to be linked with sustainable development and, thus, related future investment to be justified by environmental concerns and socio–economic needs, rather than technical advancement and industrial competitiveness (Bibri and Krogstie, 2017b).

In essence, there are two mainstream approaches to smart cities: (i) the technology- and ICT–oriented approach and (ii) the people–oriented approach. Specifically, there are smart city strategies that focus on the efficiency and advancement of hard infrastructure and technology (transport, energy, communication, waste, water, etc) through ICT and strategies that focus on the soft infrastructure and people (social and human capital in terms of knowledge, participation, equity, safety, etc) (Angelidou, 2014). Several challenges have been identified, among which to explore the notion of the city as a laboratory for innovation and to develop technologies that ensure informed participation and create shared knowledge for democratic city governance (Batty et al., 2012). As for the second approach, Neirotti et al. (2014) described smart cities as a means of enhancing the quality of life of citizens. Smart cities entail human and social factors, apart from physical and technological factors (Galán-García, Aguilera-Venegas and Rodríguez-Cielos, 2014). Lombardi et al. (2012) emphasised additional soft factors such as participation, safety and cultural heritage.

Local city governments are currently investing in advanced ICT to provide technological infrastructures supporting AmI and UbiComp, as well as to foster respect for the environmental and social responsibility (Solanas et al., 2014). In order to facilitate participation and inclusion under these premises, situated engagement might help by increasing the awareness of the participants regarding the challenges and opportunities of the urban environment where they live and work.

2.3 Learning

The process aiding citizen participation is also a process of learning, improving environmental awareness, knowledge and personal skills (Wilks and Rudner, 2013) and teaching people how to negotiate and respect each other's views (Corsi, 2002).

Smart cities are a recognised eco-system for learning. SCL aims to support the improvement of all key factors contributing to the regional competitiveness of cities: mobility, environment, people, quality of life and governance (Hollands, 2008). This approach aspires at optimising resource consumption and saving time, improving flows of people, goods and data[1]. Data produced in the city, social challenges affecting the population and technologies are all taken into account and used to increase awareness and lifelong learning through technological applications where citizens represent active actors.

2.4 Technology

Technology is a fundamental component of smart cities. Through technology, urban data can be collected, aggregated and analysed to extract valuable information. Digital solutions can also help lower the costs and increase the efficiency in many strategic sectors like governance, transportation, infrastructure and social services for the citizens. Urban data can support researchers and decision-makers in discovering the hidden layers of smart cities, highlighting patterns and processes that were otherwise impossible to detect (Vazifeh et al., 2018).

IoT plays an important role as an enabling technology for this vision. Sensor networks and M2M systems are largely employed to monitor the city and sense and collect the data generated in the urban environment. In addition to this data-driven approach, ongoing research is also exploring how technology can be used to better connect with the population, reinforcing the social aspect and building a sense of community.

[1] www.mifav.uniroma2.it/inevent/events/sclo/

2.5 SCL as a Domain Case

The domain of SCL has been defined in the early phases of my research, when drawing the boundaries for an exploratory study about the state of the art, described in P1. It is important to underline how SCL is an emerging field, a wide umbrella under which the concepts exposed in the previous sections of this chapter partially overlap. SCL addresses all the components of a smart city described above, which already represents a domain specialisation since many smart city applications found in literature do not include for example a learning or participation component. In addition, the criteria defined and employed in P1 include a social and an urban perspective (see P1 §2.1). The studies analysed in P1 and considered relevant for the survey conformed to at least one of these two criteria, meaning that on the social perspective they did not exclude any category of citizens and on the urban perspective they were referencing to an environment only to be found in cities. Given the exploratory nature of the literature mapping in P1, some studies that overlapped only partially with the definition of SCL were included in the analysis to provide a comprehensive understanding of gaps and trends by analysing a wider research field.

The philosophy of SCL and the values it promotes present many points of contact with the social and technological framing described in Chapter 1. More in particular, SCL is a relevant domain case for my work since it is connected to the following notions:

- Citizens as active actors, directly involved in and contributing to the urban environment, instead of being relegated to a passive role.
- The urban population as non-expert users, citizens often fit in this category, which represents the target user group for my research.
- The paramount role of new technologies, able to keep the user in the loop.
- The use of technology as a tool to also promote learning, inclusion and social awareness.
- The possibility of creating meaningful technological applications, tackling recognised challenges affecting most of the population.

These shared values create both an opportunity to contribute to the SCL domain, by generating new solutions and methods addressing societal challenges affecting cities, and a compelling design space to experiment with innovative technologies.

3 Related Work: IoT Toolkits

In the HCI field, Greenberg (2007) defined toolkits as a way to encapsulate interface design concepts for programmers, as a language that facilitates creation (Myers, Hudson and Pausch, 2000). More widespread in the literature is the concept of toolkits as a means to describe various types of software, hardware, design and conceptual frameworks. More articulated definitions based on the original one from Greenberg were subsequently proposed, defining toolkits as generative platforms designed to create new interactive artefacts, provide easy access to complex algorithms, enable fast prototyping of software and hardware interfaces and/or enable creative exploration of design spaces (Ledo et al., 2018). Toolkits can support users via programming or configuration environments consisting of many defined permutable building blocks, structures or primitives, with a sequencing of logical or design flow affording a *path of least resistance* (Ledo et al., 2018). Wobbrock and Kientz (2016) viewed toolkits as contributing *artefacts*, where *'new knowledge is embedded in and manifested by artefacts and the supporting materials that describe them'*.

Research on and involving toolkits is considered *constructive research*, defined as *'producing understanding about the construction of an interactive artefact for some purpose in human use of computing'* (Oulasvirta and Hornbæk, 2016). Thus, they are generative platforms designed to create new artefacts, while simplifying the authoring process and enabling creative exploration (Ledo et al., 2018).

Ledo et al. (2018) summarised the value of toolkits into five goals:

- **G1** - Reducing Authoring Time and Complexity. Concepts are encapsulated to simplify expertise, making it easier for users to author new interactive systems (Greenberg, 2007; Olsen Jr, 2007).
- **G2** - Creating Paths of Least Resistance. Defined rulesets and pathways lead the users to consistent solutions (Myers, Hudson and Pausch, 2000).
- **G3** - Empowering New Audiences. Thanks to the reduced authoring effort, new audiences can be involved in the authoring process, like artists and designers (Olsen Jr, 2007).

- **G4** - Integrating with Current Practices. Existing infrastructures and standards can be leveraged as building blocks, enabling power and robustness in combination (Olsen Jr, 2007).

- **G5** - Enabling Replication and Creative Exploration. Implementation and replication of ideas to explore a concept can be the first step to the creation of a new suite of tools (Greenberg, 2007).

Many of these goals are well adapted to tackle the research objectives presented in Section 1.3. For example, G5 promotes the creation of new tools, which can help overcome the lack of human-centred IoT solutions in the SCL domain. G2 and G3 extend the influence to new audiences, directly involving users in the authoring process and facilitating prototyping. This approach suits the co-design methodology and assists the involvement of non-expert users. The reduced authoring time described in G1 allows iterating quickly during hands-on activities, achieving tangible results in less than a day of work.

3.1 Common Toolkit Architectures

Toolkits are typically different from systems that perform a single task (e.g. an algorithm or an interaction technique) as they provide generative, open-ended authoring within a design space (Ledo et al., 2018). They support the creation of different solutions by reusing and combining the building blocks provided. These blocks can take the form of software modules with a simplified interface or hardware components that can be easily recombined or implement other forms of encapsulation and abstractions connected to specific realms like interaction modalities, communication protocols or application domains. This open-ended, generative authoring process within a domain space differentiates toolkits from systems that perform a single task. Different solutions can be created by recombining and adapting the building blocks provided, through a process significantly quicker and simpler than building a dedicated system.

3.2 Card-Based Toolkits for IoT

During co-design activities, brainstorming cards are often used to promote idea generation (Vaajakallio and Mattelmäki, 2014). Using cards makes focus change easier (Hornecker, 2010); cards can act as a mediator to the conversation between participants from different backgrounds during creative workshops (Carneiro and Li, 2011). The nature of cards inherently supports non-expert users hiding unnecessary complexities, while playing an informative role. They allow users to brainstorm and explore ideas focusing on the design rather than dealing with technical constraints. During such activities, users can point to, discuss and pass around cards, encouraging collaboration and social interaction and fostering creativity (Carneiro and Li, 2011). The outcome of this process is usually a design idea, or the framing of a small project, towards which the users develop a better level of connection and empathy.

Several card-based toolkits supporting brainstorming activities for the IoT can be found in the literature.

- IoT service kit[1] uses paper maps, 3D printed tokens and cards as artefacts representing the context, domain, assets and interactions. The toolkit supports the framing of several layers of an IoT solution. It is possible to define the needs in terms of technologies, sketch a user journey and map the flow of data.

- Mapping the IoT[2] was created to refine already existing ideas and to support the user in the process of enriching and augmenting through technology existing products. The toolkit promotes a meta-design approach, starting with the definition of the target users, markets and technologies and using these findings as a focus point to define the project brief. The cards contain elements of the design process, technology, context, strategy and interaction techniques.

- IoT design deck[3] covers the brainstorming and design phases of an IoT application through several decks of cards. As a first step, participants define the domain, the target user, the problem and which type of technology to use. Five additional decks of cards are used to inspire the design, propose provocative themes and define the inputs and outputs employed.

- IoT ideation cards[4] is a customisable card deck to conceptualise and define IoT product ideas. The cards can be arranged to compose a storyboard to illustrate logic flows, data networks and use cases. The goal is to allow any kind of team to visualise all the components that play a role in an Internet connect product.

- Know cards[5] represent simple electronic components, divided into decks representing sensors, actuators, power sources and connectivity. The cards can be used to brainstorm technology-driven applications, to learn about already existing components or to play with ideas involving random cards.

These toolkits were a source of inspiration for my work. However, none of them presented all the necessary features to cover my research objectives, described in Section 1.3. For example, some of the toolkits are meant to be used by professionals or necessitate direct supervision from design experts to be employed. Others cover only a narrow design space, whereas some of them are meant to be used through a process lasting several days or weeks.

[1] iotservicekit.com
[2] mappingtheiot.polimi.it
[3] iotdesigndeck.com
[4] sites.google.com/studiodott.be/research/iot-ideation-cards
[5] know-cards.myshopify.com

3.3 End-User Development for IoT

In the context of the work included in this thesis, end-user development (EUD) is relevant in connection with the RapIoT toolkit, described and evaluated in P6 and P7. Although relevant for the future development of the RapIoT programming paradigm (see Section 4.4), specific EUD approaches were not investigated in P6 and P7. Effort was placed in evaluating the transition between design and prototyping, while exploring diverse solutions in terms of EUD would have required effort and resources not compatible with the time at my disposal. Despite this, in the following I recall from the literature in EUD for IoT a few concepts and definitions framing EUD in the context of the RapIoT toolkit.

Cypher (1993) defined the end-user as the *'user of an application program'*, who is not a programmer but *'uses a computer as part of daily life or daily work, but is not interested in computers per se'*. Connecting this definition to the notion of IoT adopted during my work (see Section 1.1.1), the end-user is the user who is able to manipulate and interact with a smart, tangible object. EUD has been defined as *'a set of methods, techniques, and tools that allow users of software systems, who are acting as non-professional software developers, at some point to create, modify or extend a software artefact'* (Lieberman et al., 2006).

The philosophy behind the RapIoT toolkit foresees the creation of an IoT application whose behaviour is adaptable, that *'enables user-customisable behaviour'* (Trigg, Moran and Halasz, 1987). An IoT application can be adaptable in several ways. RapIoT allows to parameterise the application, offering different behaviours, as well as to interface to remote services for data exchange, by encapsulating the code into component that can be recombined (Baldwin and Clark, 2000).

In the era of IoT, EUD is not confined anymore to the software layer. End-user developers are required to acquire and apply knowledge related to electronics, sensors and actuators. A basic understanding of the relationship between the software and hardware is also required in order to solve problems that arise (Booth, Stumpf et al., 2016). Several hardware toolkits attempt to facilitate this process by simplifying the way hardware components are assembled. With RapIoT and the electronic *stickers* described in Section 4.2, we aim at removing the need to assemble any electronic circuit, while maintaining the possibility to embed sensors and actuators into an object.

4 The Tiles IoT Framework

In this chapter, the concept of IoT adopted in my research is described. The domain is defined by specifying boundaries, recalling the theories embraced and explaining how this scenario was deployed into the *Tiles IoT framework*. The Tiles framework is composed of a set of tools and a process, guiding the users towards the creation of an IoT application. The concepts described in this chapter are reflected in the research work performed during my PhD, but they also extend into future improvements and developments.

4.1 Process

The process defined by the Tiles IoT framework is designed to guide non-expert users through the creation of an IoT application. It is divided into five *phases*, reported in Fig. 4.1. The process starts with the problem elaboration phase and finishes with the production of a low-fidelity prototype of the IoT application. A brief description of the different phases is provided in the following paragraphs.

Problem

During this first phase, the users select or define a problem that they are willing to address. In order to ease the start of the process, during my work, I often provided the users with a small set of design problems connected to smart cities to choose from.

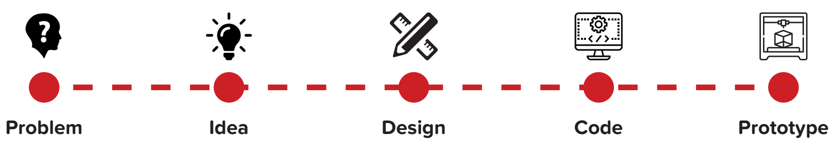


Figure 4.1: The process envisioned by the Tiles IoT framework.

Idea

During this phase, the users combine the different conceptual building blocks that compose a typical IoT application in order to generate smart objects that are relevant to the problem at stake. This brainstorming phase is fuelled by the creativity of the users, who are free to experiment and discuss different solutions and combinations of technology-augmented artefacts. In the papers included in Part II of this thesis and in the remaining of Part I, this phase is referred to as the *brainstorming* or *idea generation* phase.

Design

During the *design* phase, the smart objects devised while brainstorming are complemented with a use case scenario. This step guides the users through defining how the objects behave, deciding what the logical connections are among the components of the smart objects and clarifying how the objects interact with each other and with the end-users. The users are also encouraged to reflect on the solution created, analyse it under a different perspective and eventually modify it to match distinct quality criteria.

Code

The IoT application design, behaviour and logical connections are translated into code during this phase. I will refer to this phase as the *coding* or *programming* phase throughout the remaining of this thesis.

Prototype

The final step consists in physically assembling a low-fidelity prototype of the smart object, which will exhibit the behaviour programmed during the *coding* phase. The prototype created is intended to serve as a physical demonstrator. Efforts are made to improve the speed of development and the possibility of quickly iterating on the implementation, rather than on the complexity of the solution and the long-term stability. I will refer to this phase when discussing *physical prototyping*, *rapid prototyping* or *tangible exploration of ideas* in the next chapters.

4.2 Tools

In order to support the five phases of the Tiles IoT framework process, two toolkits were created from scratch within the Tiles project. When combined, the *Tiles ideation toolkit* and the *RapIoT toolkit* support all the phases of the process defined by the Tiles IoT framework (Fig. 4.1). The toolkits are composed of an integrated set of technologies and design tools supporting non-experts in a consistent way. The first three phases of *problem elaboration*, *brainstorming* and *design* are supported by the *Tiles ideation toolkit* (see P2, P3 and P4), whereas the last two phases of *programming* and *prototyping* are addressed by the *RapIoT toolkit* (see P6 and P7). A detailed overview of the process in relation to the toolkits is provided in Fig. 4.2.

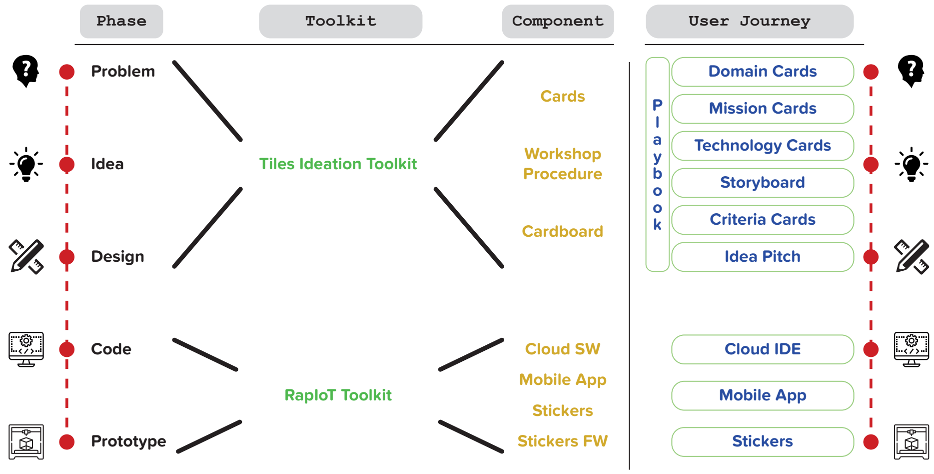


Figure 4.2: Mapping of toolkits, components and phases of the Tiles IoT framework.

The toolkits were designed to support object augmentation as a design strategy for IoT applications. This approach foresees an object as a means of interaction with the **user** the **ambient** and the **network**.

User. When manipulated by the users, augmented objects should be able to sense and distinguish different input commands or gestures (e.g. when a user knocks on, grabs, tilts or shakes an object). As an analogy, similar interaction gestures are widely known and employed in mouse-based interactions (clicking, dragging, double-clicking, etc) and touch interfaces (tapping, double-tapping, swiping, pinching, etc), but they have not been defined for generic object-based interactions. Augmented objects can also provide sensory feedback and communicate back with the users through actuators, for example, in the form of sound, light or vibration.

Ambient. Objects can sense the ambient where they are positioned and used. Distributed sensing is one of the most popular application domains of IoT. However, in the IoT framework described here, the ability to sense the ambient surrounding the object is not the main function of the object itself. Ambient data like temperature, air pollution and humidity can be used to extend the affordances exposed, enriching the user experience provided by the IoT application.

Network. Connected objects allow integration with services and data streams available on the Internet. Traditionally, IoT applications use networks also to transfer upstream sensor data collected on the field. While these are important use cases that are not discarded by the IoT framework described here, we emphasise the opportunity that a logical *mesh* network topology can offer. A single IoT application can make use of multiple objects that exchange information among themselves. Objects can either sense or provide feedback, or a combination of both. IoT can be more than a network of sensors that only communicates with a central server and presents data through a screen. Ambient sensing, user interaction and feedback are distributed in the environment and interconnected and are part of the same application logic.

4.2.1 The Tiles Ideation Toolkit

The *Tiles ideation toolkit* (Fig. 4.3) consists of several decks of cards representing the building blocks of an IoT application, a cardboard scaffolding the use of the cards and a workshop technique, which guides the users step by step in the idea generation and design process. The *Tiles ideation toolkit* presents a set of unique characteristics that are not found in already existing card-based toolkits:

- **Velocity**. It allows generating and discussing an idea in less than two hours.
- **Self-containment**. All the materials required are available to the user from the very beginning.
- **Autonomy**. Participants do not need to be supervised or guided by facilitators to complete the creative process.
- **Accessibility**. No technical skills, domain knowledge or design abilities are required.

In the following I will describe the artefacts and components that constitute the Tiles ideation toolkit. Their relations to the phases of the Tiles IoT framework are also detailed.

Components

The phases of *problem elaboration*, *brainstorming* and *design* are addressed by the Tiles ideation toolkit. The workshop technique envisioned by the toolkit is described in detail in the *playbook* visible at the bottom part of the cardboard (Fig. 4.4). The highlighted sectors on the cardboard in Fig. 4.4 can be mapped to the first three phases of the Tiles IoT framework process; the cardboard itself and the workshop technique support all the three phases. The individual components of the Tiles ideation toolkit are illustrated in the following.

Personas and ***scenarios*** cards. These cards are used during the initial *problem elaboration* phase, where workshop participants select a target user and a problem to address with their IoT application idea (Fig. 4.5).

Things, ***sensors***, ***services***, ***feedback***, ***human actions***, ***missions*** and ***connectors*** cards. Workshop participants can define one or more smart objects and model their interactions with the user by combining these decks of cards during the *brainstorming* phase.

Criteria cards, ***storyboard*** and ***idea pitch***. In the final *design* phase, workshop participants are encouraged to illustrate a use case for their idea, involving their

Figure 4.3: The last version of the Tiles ideation toolkit. The cardboard and the cards, including the extensions described in P3, are visible in the picture.

target user and the augmented objects devised and addressing the problem selected. In order to do that, they sketch a storyboard using post-it notes. The *criteria* cards are also used during this stage to further refine and specialise the idea. Finally, a short text used to pitch the idea is created, thus condensing the ultimate outcome (Fig. 4.4).

For a graphic overview of the final result in terms of creative artefacts and ideas, see the pictures included in P3.

4.2.2 The RapIoT Toolkit

The *RapIoT toolkit* supports the last two phases of the process defined by the Tiles IoT framework: *coding* and *prototyping* (Fig. 4.1). These two phases are supported by the use of programmable electronic devices for object augmentation (P7), which function and can be programmed thanks to dedicated software components (P6). Implementing rapid prototyping as object augmentation allows the users to quickly explore the concepts generated during the previous phases. The RapIoT toolkit includes both software and hardware components, allowing the creation of programmable augmented objects.

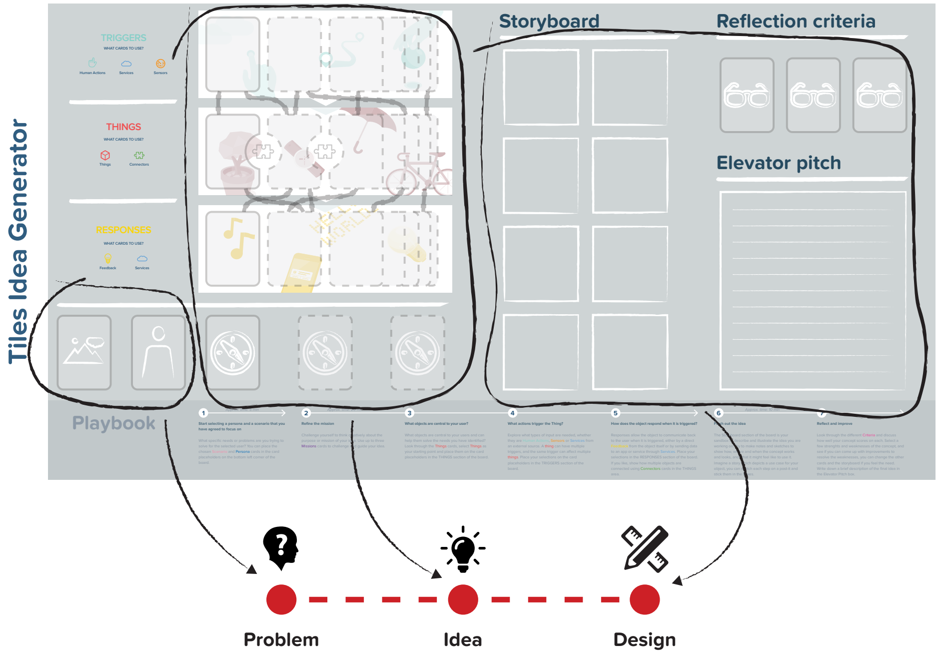


Figure 4.4: The cardboard included in the Tiles ideation toolkit. The circled sections are connected to the phases of the Tiles IoT framework process.

Hardware Components

Stickers. As in the other phases, objects remain central and are augmented through electronic *stickers*, which provide sensing and actuation capabilities in a flexible way (Fig. 4.6). Modern low-power microcontrollers can be easily embedded in a *sticker* measuring only a few centimetres in length. In a package smaller than 5 x 5 mm, the microcontroller employed includes considerable processing power, communication lines to interact with the attached sensors and actuators and support for secure, IP-based wireless connectivity. These microcontrollers can operate on batteries for years, taking advantage of low-power sleep modes when idle. The *stickers* are battery-powered and connected to the cloud through a gateway, or directly without the need for any other device. Different types of *stickers* can provide different combinations of I/O; many of them can potentially be attached to a single object. They can trigger events while sensing the surrounding ambient or when the users interact with the object that they are attached to. The data packet representing the event is transmitted to the cloud, where the application logic is running. The *stickers* can also consume events received from the cloud (e.g. a command to vibrate). The type and number of events that can be consumed or generated by a *sticker* are only dependent on the type and number of sensors and actuators it has onboard.

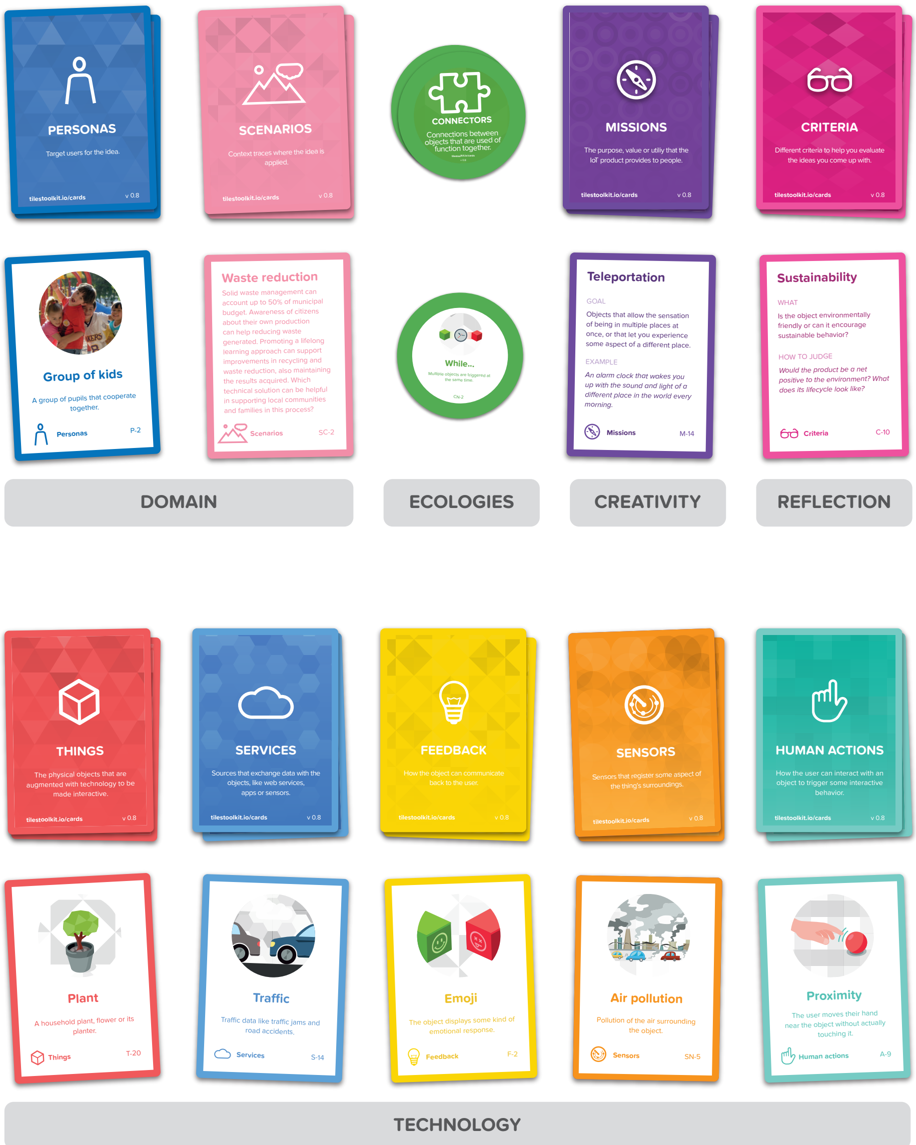


Figure 4.5: Cards included in the Tiles ideation toolkit.

Figure 4.6: Electronic *stickers* employed by the users to prototype the smart objects.

Software Components

In terms of software, we distinguish two loosely coupled sub-domains: (i) the integrated development environment (IDE) that allows the users to program the application logic and (ii) the platform software stack running on the *stickers*, the cloud and eventually the gateway.

Development Environment. The development environment is a cloud-based ambient where the users can program the IoT application logic. The editor used to write the code runs in a browser, sparing the users the installation of any toolchain, driver or software on their personal devices. The programming paradigm employed is a simplified domain-specific language (DSL) based on JavaScript.

Stickers Platform Software Stack. The software running on the microcontroller contained in the *stickers* is tasked to fetch data from the sensors attached, command the actuators and expose an API on the network interface. Through the API, the *stickers* can be remotely controlled and can also notify the cloud if new sensor data are available or if any user interaction occurs.

Cloud Platform Software Stack. The cloud software acts as a network hub for the *stickers*, exposing their functionalities to the development environment. The development environment, accessible from the browser, is also hosted on the cloud.

The cloud IDE is the only interface for the user to program the application logic. To start coding, the user is not required at all to connect any cable or install any software.

Gateway Platform Software Stack. The gateway is a device whose main role is to bridge the connectivity between the *stickers* and the cloud. For example, it can provide Bluetooth connectivity to the *stickers* on one side and cellular connectivity on the other, allowing the *stickers* to send and receive events to and from the cloud. We used a smartphone to act as a gateway, and a multi-platform mobile app was developed to handle the connectivity towards the cloud and the *stickers*.

Using IP-based wireless connectivity on the *stickers*, it is possible to reduce the complexity of the gateway, since it would simply need to forward the IP packets between the network interfaces. This way, the gateway would work at a lower level of the ISO/OSI network stack, thus requiring fewer computational resources compared to a routing task involving the full translation of application-level protocols. The use of a gateway can possibly be omitted altogether using new low-power microcontrollers with embedded cellular connectivity. Thanks to these solutions, the *stickers* can have direct access to the cloud like a smartphone.

4.3 Integration of the Toolkits in Prototyping

With the Tiles ideation toolkit and the RapIoT toolkit, we introduced a path of continuity between the phases of *problem definition*, *idea generation* and *design* and the creation of a prototype for quick and tangible exploration of creative ideas.

Connecting these two stages, without limiting the outcome to a simple idea, means supporting the users in the conceptual as well as in the practical stage. This continuum is important for several reasons, as explained by Brown (2005):

- Ideas presented using only words are highly open to interpretation, and require supremely engaging storytelling skills to be relied upon effectively. Words mean different things to different people, especially if they come from different backgrounds.
- A prototype, or a demonstrator of the idea, describes a concept in a way that is not open to many interpretations. Physically experiencing the nature of a prototype facilitates convergence towards a shared view.
- A prototype is simultaneously an evaluative process (it generates feedback and enables corrections) and a storytelling instrument (it visually describes a strategy or an idea).

The goal is not to create a close approximation of the finished product, but to build something that is rough and ready and works, an instrument to elicit feedback, to unlock intuitions from the people (T. Brown, 2005).

The Tiles IoT framework guides the users through the different phases in a consistent way. The theoretical concepts, the building blocks of the toolkits and the abstractions employed are introduced gradually and progressively enriched during each phase. This steady learning curve allows capitalising the efforts towards the prototype, avoiding drastic shifts of paradigm or introducing gaps in the knowledge required, which might hinder the creative process.

The connection between the Tiles ideation toolkit and RapIoT is guaranteed mainly by the design of the data communication protocol of RapIoT, based on input/output primitives (see P6 §3 for details). These developer-friendly, human-readable constructs represent the data packets exchanged between the application logic in the cloud and the electronic *stickers*. They encapsulate events and commands being transferred on the network. The primitives, that are directly employed by the end-user when programming the application logic, can be mapped to the cards included in the Tiles ideation toolkit (P7 §3). In Fig. 4.6, an example of this path of continuity is discernible: the sensors cards for temperature and humidity are also implemented as RapIoT primitives, these primitives are generated by the *stickers* and their data payload (e.g. the temperature in °C) is managed directly by the end-user who programs the application logic.

4.4 Lowering the Bar: Non-Experts as Target Users

The Tiles ideation toolkit and the RapIoT toolkit were designed to keep the entry barriers low, at the same time allowing the users to generate an idea, illustrate it and create a prototype during workshops lasting less than a day. The target group is represented by *non-expert* users, described in Section 1.1.2, who have some proficiency in the basic paradigms used by high-level programming languages such as JavaScript and Python. Upper secondary school students often belong to this category.

The holistic approach of the Tiles IoT framework allows non-experts to actively contribute during tasks that have traditionally required specific technical skills, like low-level programming, assembly and prototyping of electronics for augmented objects.

Since the RapIoT toolkit targets low-complexity applications and non-expert users, it might be beneficial to employ a programming abstraction that requires a lower level of programming skills than the ones usually required by textual programming languages. Visual programming languages like Scratch[1] or Node-RED[2] have been successfully employed in rapid prototyping, and their steep learning curve is appreciated by novices (Booth and Stumpf, 2013). These well-established programming paradigms are easy to use and are understandable interfaces for the configuration of IoT devices (Houben et al., 2016).

[1]https://scratch.mit.edu/
[2]https://nodered.org/

Several factors led to the choice of having non-experts as target user group:

- **Extended Reach**: We wanted to empower new, wider audiences in design and prototyping for IoT. The majority of the population does not possess advanced skills in programming, electronics or design (P6, Table 3).
- **Research Gap**: Existing design and prototyping toolkits for IoT often require specific skills during some of the steps needed to transition from problem definition to a low fidelity prototype (P6 §1).
- **Matching the Application Domain**: Co-design in the SCL domain foresees the involvement of citizens, who often fall into the definition of non-experts.

4.5 Identified Values of the Toolkits

Here, the generic values of the toolkits, identified by Ledo et al. (2018) and reported in Chapter 3, are recalled and contextualised for the toolkits described in this chapter.

- **G1**: Reducing the Authoring Time and Complexity. The toolkits presented are designed to be used during activities lasting less than a day. Complexity is kept under control by gradually introducing new concepts along the process.
- **G2**: Creating Paths of Least Resistance. The defined rulesets and pathways are present in the toolkits in the form of step-by-step instructions on the playbook and through constraints guiding the coding and prototyping activities.
- **G3**: Empowering New Audiences. The target group of non-expert users, defined as in Section 1.1.2, are fully involved in the authoring process and represent audiences not traditionally addressed by IoT toolkits.
- **G4**. Integrating with Current Practices. The use of cards and other building blocks that are already familiar to many users facilitates development and integration.
- **G5**. Enabling Replication and Creative Exploration. Ideation, prototyping and replication of creative ideas are used to explore new concepts and tools.

4.6 Learning from the Past: Tackling Current IoT Challenges

We are all familiar with what the World Wide Web is and how it works. It is nowadays an essential part of our daily routine, work and leisure. However, we cannot say the same about IoT, despite it being a concept as old as the web. In order to provide the reader with a provocative perspective of the IoT field in its current status, in the following, I will describe the web as if it shares the shortcomings and limitations of the current IoT technologies.

> Alice just bought a new computer, produced by *BigTechCompany*. She also bought from the same company a cable modem, which will allow her new computer to connect to the telephone line and access the Internet. Using the browser that she found pre-installed on her computer, she quickly discovered that she can only access a limited number of websites, which are associated with or supported by *BigTechCompany*. In order to visit some of the other websites, she needs to install a second browser on her computer, but the website selection will still be quite limited. She wonders how she can connect to the websites associated with *SmallTechCompany*. It seems that a few of the websites of *SmallTechCompany* can be accessed by replacing her cable modem with one produced by *SmallTechCompany*. However, to access all of the websites of *SmallTechCompany*, there is no other way than buying a new computer produced by *SmallTechCompany* itself, which is also the only computer able to run the browser issued by the company. She decides to buy the required hardware, even if it looks like a waste of money given that she already has a brand-new computer. After a few months, to her great disappointment, *SmallTechCompany* goes out of business, leaving Alice with an expensive computer-shaped paperweight: the websites of *SmallTechCompany* are taken offline, the browser cannot access any other content and the modem does not even connect anymore to the Internet. Alice hopes for someone to take over the work of *SmallTechCompany*, but she gets to know that *SmallTechCompany* did not release any documentation or specification covering the technologies used. That knowledge was condemned to disappear with the company.

It is difficult to imagine how this fictional scenario could support widespread adoption, growth, accessibility and security. For example, adoption is inhibited by proprietary specifications, accessible only by payment. Growth and accessibility are limited if only a few market players have the resources to develop the technology. Security is affected since it will not be guaranteed if the company licensing the technology goes out of business, leaving the devices of the users exposed *in the wild*.

Many researchers and IoT experts have precisely identified these issues and proposed solutions that build on the aspects that made the web so popular and so different from what described above. This approach was adopted by Guinard and Trifa (2016) and was described in their *web of things* architecture. They advocated that the simplicity and openness of the web and its standards are likely what enabled the web we know today. The lingua franca of the web enabled the users to access any webpage without installing anything and has been a major factor in its success. By enabling webpages, browsers, servers and services to all speak the same application language, the integration of a large variety of content was incredibly simplified. Unfortunately, no equivalent enabler has yet been found for devices and applications in IoT (Guinard and Trifa, 2016, p. 23).

Their pioneering work was not limited to a theoretical exploration. A W3C working group has been established[3] and is currently active in drafting an open standard based on the *web of things* model. A preliminary implementation of the standard is already in use by the Mozilla IoT gateway[4], an open-source hub for home automation.

Many other standards, protocols and data models are available in the IoT world. To cite a few examples, the Bluetooth GATT profiles[5], the Open Connectivity Foundation oneIoTa[6] and the Zigbee Dotdot[7] all define a mix of APIs, data models and communication protocols to be used with different *things*. Similar ecosystems have been designed and implemented by Apple, Google and Amazon. These standards are often not as open and accessible as the web ones, and they fall short when addressing IoT under a user-centred perspective. For example, they define APIs to communicate with objects, to allow data sensing and collection, but none of them have yet addressed human interaction primitives and gestures used to manipulate objects, using structured, open and accessible specifications.

4.7 Vision for the Tiles IoT Framework

With the Tiles project, we experimented with and evaluated a user-centred approach to IoT based on physical objects and tangible interfaces. In order to support the target user group of non-experts, our aim was to simplify the setup and deployment tasks of a typical IoT application, providing an intuitive development environment that allows rapid prototyping and fast exploration of ideas. These goals very well fit with the philosophy of the *web of things*, which in addition enables community work, transparency, openness and accessibility. Building future advancements on top of this model will prevent vendor locks and expensive implementations and will allow the Tiles *stickers* to be employed for multiple use cases and by multiple actors, with minimal integration costs even when detached from the RapIoT platform.

Outreach, education and open collaboration are the main principles driving the work of W3C. The adoption of a model backed by an international standards organisation such as the W3C guarantees a solution that is not tied to the fate of a commercial corporation and is royalty-free to use for anybody.

[3] https://www.w3.org/WoT/WG/
[4] https://iot.mozilla.org/
[5] https://www.bluetooth.com/specifications/gatt
[6] https://www.oneiota.org/
[7] https://www.speakdotdot.com/

5 Theoretical Underpinning

In this chapter, the predominant theories defining the approach adopted by the Tiles toolkit and the IoT ideation process will be presented. It is worth mentioning that the investigation work that led to the creation and evaluation of the Tiles toolkit and the IoT ideation process was grounded in additional theories such as UCD, EUD, SCL and CSRL and took advantage of the HCI guidelines, for example, in connection to usability. Some of these theoretical domains have been addressed in Chapter 2 and Chapter 3, whereas others will be covered in Chapter 6.

5.1 Co-Design

In the context of sustainable HCI, distinctions were drawn between seeing the behaviour of the users as the cause of environmental problems and gathering needs and opportunities from users to inform design (DiSalvo, Sengers and Brynjarsdóttir, 2010). By means of participation and co-design, people can provide direct input to *solve the problem of the users*, designing thus their own behaviour change (Lockton et al., 2014). Through co-design, users build empathy with the solution they contributed to. Co-design allows multiple voices to be heard; it is fair and ethical to involve those whose livelihoods, environments and lives are at stake in the decisions that affect them (Perlgut and Sarkissian, 2005). Employing participatory approaches empowers people, allowing shared responsibilities among stakeholders. This can create credibility and trust, emphasise diversity in stakeholder involvement and increase the likelihood that the final product will meet the expectations (Pettersen and Boks, 2008). The role of the researchers is to support, explain, fight and help negotiate design tensions, recommending methods and tools. Controlling the creative process and managing the users are practices that do not belong to the co-design methodology.

Under the umbrella of co-design, various approaches and methods are used, which result in different combinations between the level of involvement of the stakeholders and the number of different stages of the design process covered. Many studies that apply co-design in the smart city context involve the users during the brainstorming and idea generation phase (Val Mitchell et al., 2016; Schuurman et al., 2012; Mechant et al., 2012; Fu and Lin, 2014), whereas others consider the shareholders only in a tokenistic way, limiting their contribution to the act of providing simple feedback or rating the final solution (Reiersølmoen, Gianni and Divitini, 2017). Although participation is historically emphasised at the moment of idea generation (Cross, 1971), modern co-design practices can extend beyond the very first phases of the design process, including the users more deeply throughout the whole process. This is facilitated by the lower entry barriers achievable while using recent technologies to address specific tasks, which have been a prerogative of professionals only a few years ago.

In this thesis, I will describe how co-design practices have been used to drive brainstorming and idea generation and also how co-design can be extended beyond its traditional scope. This aspect characterises how the toolkits described in Chapter 4 are intended to be used *in the wild*. The objective is to promote a co-design process where potential stakeholders are brainstorming, designing and prototyping a solution for a problem that affects them. This opportunity has been explored in few studies, which is demanding given the social and technological challenges that come attached to practices like design validation with the user and collaborative implementation of ideas.

5.2 Tangible User Interaction with Smart Objects

The work described in this thesis embraces the understanding of IoT and ubiquitous computing proposed by Rogers (2006), which challenges the original calm computing vision of Weiser (1991). Weiser depicted a scenario where user interaction is anticipated and predicted by a sensing smart environment, relegating the user to a mostly passive role. This approach led to poor results in research and prototyping on ubiquitous computing and IoT, mainly because trying to predict human will is a difficult artificial intelligence problem (Rogers, 2006). Even small errors are perceived as annoying and frustrating from the user, to the point of dismissing and abandoning the technology in use.

Such vision presents challenges also from a sociological point of view. Users should be encouraged to interact and make critical decisions; the technology can support this process, improving awareness, stimulating reflection and possibly engaging the user in the activity. This way, it is possible to introduce new techniques to change the attitudes and behaviours of people, on the basis of social learning, for sustainable common objectives.

Rogers did not completely discard the vision of Weiser but rather tried to entertain other possibilities besides calmness for research on ubiquitous computing. Examples include extending and supporting personal, cognitive and social processes such as habit-changing, problem-solving, creating, analysing, learning or performing a skill (Rogers, 2006). Rogers advocated that research in the field should be aimed at better understanding human activity rather than trying to predict and intervene in situations that already work reasonably well, with a high risk of unwanted or unpredictable outcomes. This people-oriented perspective was also envisioned by Streitz et al. (2005) in the smart object research domain. They prospected smart objects as empowering artefacts for decision-making, supporting mature and responsible actions.

In order to tackle the challenges described in Section 1.3, I chose to adopt an *object augmentation* strategy in order to create TUIs based on smart objects. As described by Kuniavsky (2010, p. 254), object augmentation starts from everyday, non-digital objects and augments them with technology, while their purpose and familiar characteristics are maintained. This family of interfaces, also called sensing-based interfaces, allows for new interaction paradigms that explore the opportunities lying beyond traditional human-computer interfaces such as screens, keyboards and mice (Van Dam, 1997). TUIs are characterised by the embodiment of interaction in physical objects, they emphasise the physicality of interaction through the coupling of physical and digital representations (Markova, Wilson and Stumpf, 2012). TUIs take advantage of the physical skills of the users, exploiting knowledge of the everyday, non-digital objects (Jacob et al., 2008). These physical interfaces are capable of delivering relevant information at appropriate times, which is critical for triggering learning and sustaining reflection (Rogers and Muller, 2006). In addition, they can be embedded into the environment and represent data streams in a physical way (Hornecker and Buur, 2006). User interaction then moves *off the screen*, becomes more natural and can be distributed in space (Dourish, 2004).

Advocating the well-known design maxim *'jack of all trades, master of none'*, Lindwell et al. (2010, p. 102) presented the trade-off between *flexibility* and *usability* in design: flexible designs can perform more functions than specialised ones do, but they perform these functions less efficiently (Fig. 5.1). Jacob (2008) transferred this concept to TUIs, proposing a trade-off between *reality* and *versatility*. TUIs are specialised by their physical affordances and constraints and, thus, can perform a limited set of tasks with a high level of realism or simplicity. Moreover, TUIs foster collaboration (Rogers and Rodden, 2003), in which they increase the visibility of the actions of others and allow for concurrent interactions. TUIs extend the static representation of an object with an intangible, dynamic one. TUIs were part of the original vision of Weiser and have been then researched by Rogers as well as Ullmer and Ishii (2000), among others.

The process defined by the Tiles IoT framework (Chapter 4) is centred around the concept of *object augmentation* as design strategy, as defined by Michael Beigl and his colleagues (2001) and reported in Section 1.1.1. However, several elements of the Tiles IoT framework extend beyond such definition. Beigl foresees smart objects as objects that are able to *'exchange information about themselves with other artefacts and/or computer applications'*, while the Tiles ideation toolkit and RapIoT include additional affordances, typical of TUIs. For example, *human action* and *feedback* cards (Fig. 4.5) keep humans in the loop, promoting the design of interaction modalities popular in the field of TUIs (P2, p. 591). The presence of TUIs elements in the smart objects designed by the users during the workshops was one of the aspects that was object of evaluation (see P3 §4.3). We assessed if the smart objects envisioned were simple network-connected sensor probes or if they involved human-interaction and/or tangible manipulation, namely if they were able to sense and react to direct human interaction. The Tiles ideation toolkit has demonstrated to successfully support the design of these TUIs based on smart-objects (P3), while RapIoT enabled the transition into a low-fidelity prototype (P7).

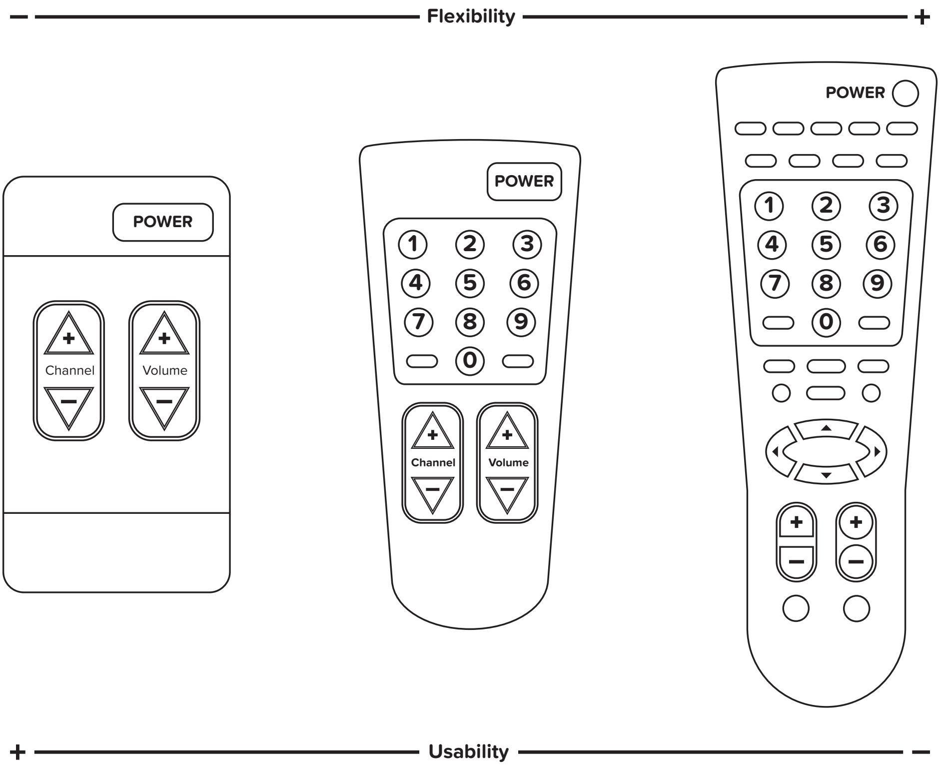


Figure 5.1: A visual example of how a simple design can be easy to use, whereas a more flexible and feature-rich one is less usable (Lidwell, Holden and Butler, 2010, p. 103).

6 Research Methodology

The aim of this chapter is to present the research methodology adopted during the work described in this thesis. Not all the methods have been explicitly elaborated in the papers included in Part II, but they have been nevertheless important while framing the studies and designing the tools employed during my research.

6.1 Research Philosophy

The leading research philosophy adopted was *phenomenology*, a variation of *interpretivism*. According to the interpretivist approach, it is important for the researcher as a social actor to appreciate differences between people (Saunders, Lewis and Thornhill, 2009). Studies adopting the interpretivism research philosophy usually focus on meaning and may employ multiple methods in order to reflect different aspects of the issue. During my research, quantitative data were collected and used to validate theories and outcomes; however, we could not refrain from considering subjective human interests and meanings to validate the results.

The *phenomenology* branch describes the philosophical approach asserting that what is directly perceived and felt is considered more reliable than explanations or interpretations in communication (Remenyi et al., 1998). Ideas and theories are generated from a rich amount of data mainly by means of induction, whereas stakeholder perspective may have its reflection on the study.

6.2 Design as Research

The topics proposed in Chapter 1 have been researched using design science research and researching design methods (Hevner and Chatterjee, 2010). Design as Research encompasses the idea that performing innovative design that results in clear contributions to the knowledge base constitutes research (Hevner and Chatterjee, 2010). Knowledge generated via design can take several forms, including constructs, models, methods and instances (March and Smith, 1995). Design as Research thus provides an important strand of research that values research outcomes that focus on the improvement of an artefact in a specific domain as the primary research concern, and then it seeks a broader, more general understanding of theories and

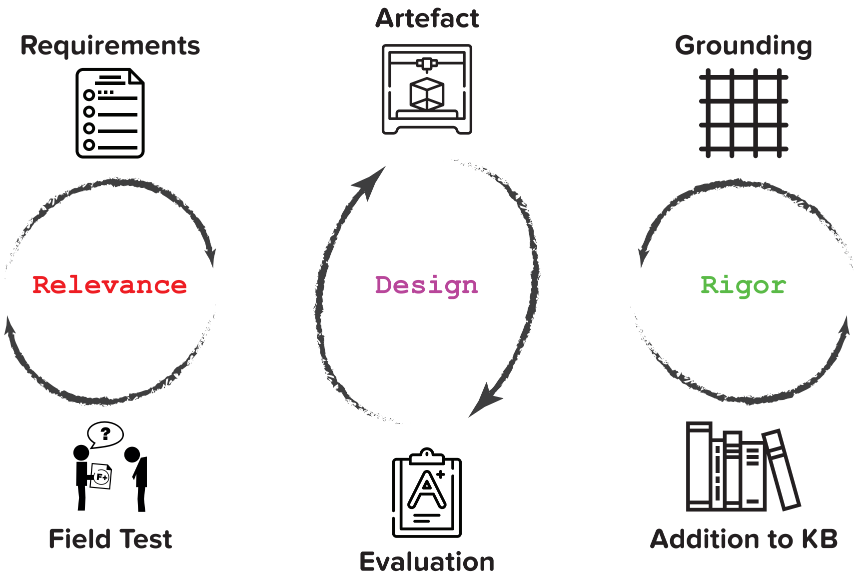


Figure 6.1: Schema of the three design science research cycles, based on Hevner's model (2007).

phenomena surrounding the artefact as an extended outcome (Hevner and Chatterjee, 2010). A key insight into how to perform design science research can be gained by understanding the existence of three design science research cycles in any design research project, summarised in Fig. 6.1 and further detailed by Hevner and Chatterjee (2007).

6.2.1 Relevance Cycle

The relevance cycle initiates design science research with an application context that not only provides the requirements for the research (e.g. the opportunity/problem to be addressed) as inputs but also defines the acceptance criteria for the ultimate evaluation of the research results (Hevner and Chatterjee, 2010). This process connects design science research with the application context. The iteration allows for improvements and refinements of the requirements and for validating incremental results using field testing feedback.

During my work, a user-centred approach characterised the work pertaining to this cycle. Data collected during user studies contributed to the process of constant refinement and improvement of the tools and technologies designed.

6.2.2 Rigor Cycle

The rigor cycle ensures that research is grounded in relevant literature and has solid foundations. This includes considering the state of the art, past knowledge is in fact essential to drive and support innovative research. A literature mapping on topics relevant to my research was conducted as part of this cycle. This represented the starting point, building a solid knowledge foundation and theoretical background for subsequent research. Design science research can also contribute to improving the state of the art and the knowledge base. This way, the rigor cycle loop is then closed, providing a tangible contribution to the field of research.

6.2.3 Design Cycle

The internal design cycle is the heart of any design science research project. This cycle of research activities iterates more rapidly among the construction of an artefact, its evaluation and the subsequent feedback to refine the design further (Hevner and Chatterjee, 2010). It is important to note the difference between the design cycle and the relevance cycle. The first iterates and validates the artefact against the requirements, whereas during the second the objects of the iteration and refinement are the requirements themselves, not the artefact. The design cycle should be well balanced between the building and evaluation processes. Scarce commitment in one of the two phases will lead to poor overall results. The design cycle is quite independent of the relevance and rigor cycles, and it is also executed more.

6.3 Research Strategy and Methodological Choice

In order to implement the main research strategy, several methods have been adopted. A mix of qualitative and quantitative research methods have been used to account for the unpredictability in field studies (Rogers, Connelly et al., 2007). Mixed methods research fit well with the research objectives because of its potential with respect to understanding and explaining complex organisational and social phenomena (Cao et al., 2006; Mingers, 2001). Further, mixed methods research has received much attention in the social and behavioural sciences recently (for a review, see Tashakkori and Creswell, 2008). During my research inquiry, observations, notes and video and audio recordings were the primary means used to collect data during the field studies with the users (Fig. 6.2). Pre-post questionnaires, interviews, quiz games and focus groups were usually employed to evaluate the artefacts employed and the process adopted and to assess the perceived outcome of the user experience.

6.4 Research Activities

During the progress of my research, different activities and outcomes contributed to the three cycles of the 'Design as Research' methodology: *relevance*, *design* and *rigor* cycles.

Figure 6.2: Video recording of the user interaction occurring around the Tiles cards and cardboard. The brainstorming workshop pictured was one of the studies included in P4.

As a first step, I performed a literature mapping of the technological applications in smart cities (P1). The identified needs, challenges and research opportunities were used to drive and support the following steps. Early design ideas were turned into exploratory low-fidelity prototypes and tested on the field with the users. Subsequent iterations both improved and specialised the prototypes, building on the experience matured during the field studies and the domain knowledge acquired while mapping the literature.

The central course of my research concentrated on the user evaluation of different prototypes targeting the *idea generation* and *design* phases of the Tiles IoT framework process. Other activities involved the design, creation and evaluation of technological tools connecting and extending the *design* phase into *rapid prototyping*. Fewer iterations were performed on such tools because of the increased complexity of the manufacturing process. However, valuable insights, prototypes, design recommendations and knowledge resulted as an outcome. Working prototypes were also used to validate and extend theories as part of the *rigor cycle*. Research outcomes were reported in academic publications (Chapter 7) and research contributions (Chapter 8) emerged. Throughout the process, the literature on co-design and tangible interaction (Chapter 5) informed the design work.

Figure 6.3: A design workshop with university students, part of the studies included in P3.

6.4.1 User Studies

The primary investigation method selected to understand the domain, familiarise with the target users and evaluate the artefacts produced during the design cycle was *design workshops* (Fig. 6.3). These workshops were used both to validate the tools employed and to inform the next design iteration. Insights pertaining to possible improvements or defects were gained from direct experience in the field and extracted from the data collected during the workshops.

The typical setting for the majority of the studies was a design workshop where participants worked in groups of two to six people. The objective for most of the workshops was to generate a design idea for an IoT application, adapted to solving a particular problem for a specific end-user. The IoT application idea included the use of smart objects designed by the participants during the workshop. During some of the workshops, the participants continued after this initial idea generation phase, physically building the smart objects embedding sensors and actuators. These smart objects were then programmed by the participants to expose high-level functions to the end-users.

The workshop participants were usually students from secondary school up to university level, aged between 13 and 27. Several workshops were also conducted with other categories of users, including researchers, IT professionals, urban planners, teachers and municipality employees. More than 25 workshops were run between August 2016 and April 2018, with more than 500 participants involved.

Observations, researcher notes and questionnaires were the primary means to collect data. In addition, video recordings, interviews and pictures of the produced artefacts were collected during some of the workshops. My role during the workshops was to present the activities and introduce the concepts of IoT and augmented objects. Later on, during the creative phase, my main task was to observe the work of the group, without intervening. The participants occasionally asked for clarifications or support, in which case I was available to provide the required help.

Data collected was analysed with mixed research methods (Venkatesh, S. Brown and Bala, 2013). The focus of the analysis was twofold. At first, it allowed validating the perceived usefulness, acceptance and learning outcome of the tools used. The outcome of the data analysis also validated the modifications made to the tools, closing the *design cycle*. The second purpose of the analysis was to spot any usability issue, either in the tools or in the process adopted. Such issues might include confusing guidelines, inappropriate terminology or unclear purpose of the tools at use. These outcomes fed the subsequent iteration of the *design cycle*.

6.4.2 Design Iterations

The work on design and prototyping was driven by the theories adopted and the requirements refined during the user studies. The design process followed a UCD approach (Maguire, 2001; Gulliksen et al., 2003). A total of eight prototype iterations were completed. Table 6.1 shows an overview of the prototypes created and the technologies used during development in relation to the papers that describe the work.

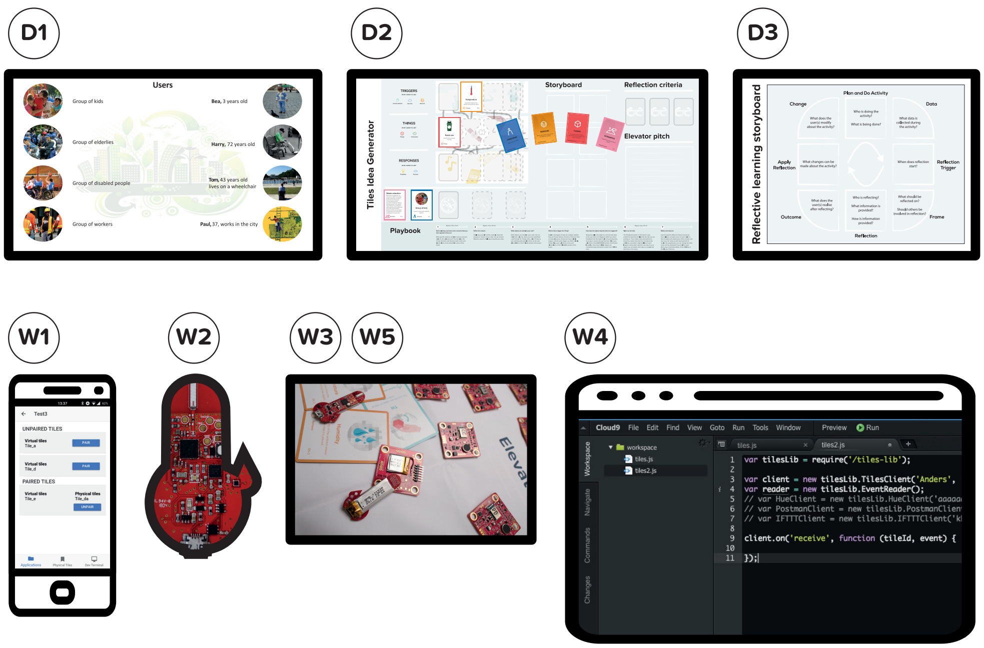
In order to support the idea generation and prototyping journey, several tools were developed during the Tiles project:

1. Several decks of cards, a tabletop cardboard to scaffold the use of the cards and a workshop protocol to guide the process, supporting the problem definition, brainstorming and design phases.
2. A cloud-based software back-end and development environment to programme the IoT application logic.
3. Two Bluetooth electronic devices embedding sensors and actuators, supporting rapid prototyping for IoT through object augmentation.
4. A multi-platform mobile app to facilitate the deployment of the prototypes, serving as a gateway connecting the Bluetooth devices to the cloud software platform.

I contributed to different levels in the design, formalisation of the process, development and scientific evaluation of the components described above, in collaboration with other students and researchers involved in the Tiles project.

Table 6.1: List of prototypes built

ID Ver.	Name	Released	Prototyping tools	Development: Software	Development: Hardware	Development: Material	Papers
D1	Tiles	Aug-16	Personas, Scenarios			•	P2
D2	Tiles SC	Jan-17	Cards, Cardboard			•	P3, P5
D3	Tiles Ref	May-17	Storyboard, Personas			•	P4
W1	RapIoT App	Jun-17	Android/iOS framework	•			P6
W2	Tiles Temp v1	Feb-18	Arduino, Electronics	•	•		Internal
W3	Tiles Temp v2	Mar-18	Electronics		•		P7
W4	RapIoT Cloud	Apr-18	Javascript Node.js	•			P7
W5	Tiles Square	Apr-18	Arduino Firmware	•			P7



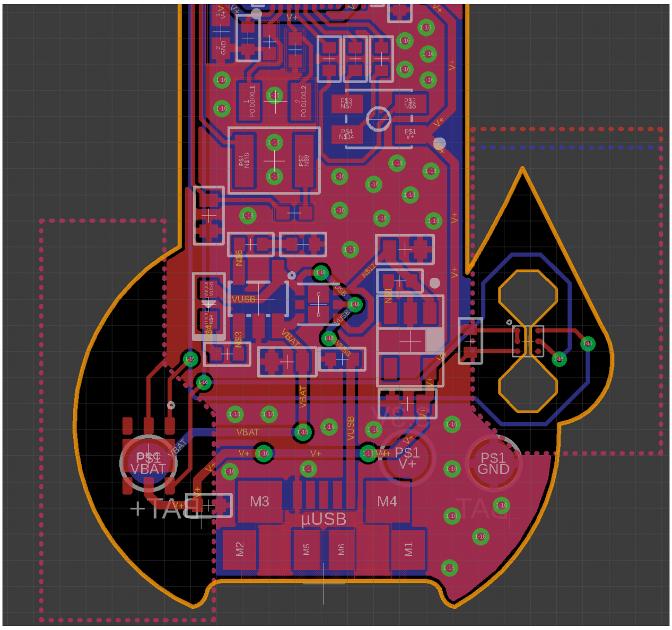


Figure 6.4: Details of the CAD circuit board of the Tiles Temp v2. The temperature and humidity sensors are located in the two dashed areas.

Building the prototypes involved a mix of design, software, hardware and material development. Software was written to create the cloud-based development environment and back-end, the mobile application, and to programme the microcontroller embedded in the electronic devices. The hardware development included the design, manufacturing and testing of electronic boards with embedded sensors and actuators (Fig. 6.4). The cards and the tabletop board were designed using desktop publishing and vector graphic editor software. The cards were printed on standard paper, on cardboard and finally on professional-grade playing-card material. The tabletop board was printed on paper, cardboard and various textile materials.

Design iterations were usually quick. As soon as incremental improvements and feedback were gathered, a new prototype was produced and tested on the field. Some of the software tools employed included Arduino[1] for microcontroller development, the Adobe Creative Cloud suite to design the cards and the cardboard and the Ionic framework to create the mobile application. Traditional and rapid prototyping approaches complemented the work. The electronics were designed in-house using the Eagle CAD software and built by an external electronics manufacturing company. An external company also printed the cards and the tabletop board in their final version.

After each iteration on the prototypes, an evaluation during a workshop was

[1] https://arduino.cc

performed. User testing allowed for maintaining a user-centred design perspective, to introduce new ideas into the process, fix defects and validate the new changes. Some prototypes were built only for internal testing purposes (W2). The Tiles cards, cardboard and workshop process were also part of an expert evaluation in October 2018, where 15 professional designers and IoT researchers from all around Europe provided feedback and suggestions after experimenting with the toolkit during a brainstorming workshop.

6.5 Research Approach in Field Studies

The research approach employed during the investigation described in this thesis is here evaluated and contextualised. A research strategy based on case studies was adopted. Case studies focus on understanding the dynamics present within single settings (Eisenhardt, 1989). During my research, this research strategy was supported by a methodology based on mixed methods, as motivated in Section 1.5. Validity and reliability issues within case studies (Yin, 2017; Riege, 2003) are also discussed.

Criteria were established for judging the quality of the case study strategy (Riege, 2003). Several tests and techniques were synthesised for establishing validity and reliability in case study research, as well as the validity of the mixed methods methodology. In the following section, the research approach is evaluated following the guidelines and design tests reported by Riege (2003); however, not all the tests devised by Riege apply to the studies in the articles included in this thesis.

6.5.1 Construct Validity

Construct validity evaluates whether appropriate operational measures have been adopted for the theoretical concepts being researched. Collecting data using multiple data sources increases construct validity and protects against researcher bias (Peräkylä, 1998; Flick, 1992). Converging findings emerged when analysing different sources through triangulation. This has been the case for the data collected during the evaluation workshops reported in all the articles except P1, where there was no workshop-based empirical evaluation. We collected quantitative and qualitative data through questionnaires, structured interviews, observations, video and audio recordings and analysis of the artefacts produced. Chains of evidence in the data (Griggs, 1987; Hirschman, 1986) were highlighted when summarising the outcome of the data analysis process. For example, in line with the iterative nature of design as research, evidence collected during the first design iteration in P2 was used to ground and improve tools and processes employed in P3, P4 and P5.

The total number of participants involved in the studies added up to more than 500. A fraction of them were involved in the studies presented in this thesis. Significant experience was gathered in the process, such experience contributed to the definition of theories and in providing insights while analysing qualitative and quantitative data. The size of the data sample allowed for some statistical analysis, but I looked for confirmation in the qualitative data before formalising the results. Given the number of human factors involved, this strategy was demonstrated to be more robust and allowed for better interpretation of the data collected. As an example, some of the dynamics of the study reported in P3 were not observable from a qualitative point of view, while an aggregated quantitative analysis didn't match with the evidence collected through observations. A more in depth analysis on the quantitative data was needed to formulate a plausible explanation of these dynamics.

During the studies, researchers had close and direct personal contact with the organisations and users involved. Effort was then made to refrain from subjective judgements during the periods of research design and data collection, to enhance construct validity. For this reason, results were discussed among the entire research group before formalising theories and constructs. Researchers not directly involved in specific studies were included in the discussion, to provide additional perspective.

Each of the studies was limited to one or two sessions, lasting usually two to three hours each. It was, therefore, not possible to prove long-term effects for the tools developed.

6.5.2 Internal Validity

Internal validity, as traditionally known in quantitative research, refers to the establishment of cause-and-effect relationships, while the emphasis on constructing an internally valid research process in case study research lies in establishing phenomena in a credible way (Riege, 2003). Researchers should not only highlight major patterns of similarities and differences between the experience of the respondents and their beliefs but also try to identify what components are significant for those examined patterns and what mechanisms produced them. An example of the need to assess the significance of these patterns is provided by the data collected during the studies of P2 and P3. In both cases we recorded discording opinions about the desired level of constraints of the design tools employed. Some of the users desired more freedom during the brainstorming session, while others felt overwhelmed by the number of choices. However, during the studies of P2 the pattern was significantly tending towards the need to have more constraints and guidance, while during the studies of P3 the pattern was not significantly pointing to any of the two cases.

Data from the field studies included in the papers presented in this thesis were checked cross-case to assess the internal coherence of findings (Miles and Huberman, 1994). Illustrations and diagrams eased this task, allowing the identification and evaluation of evidence, cross-checking within-case and cross-case (Yin, 2017).

During field studies, it was sometimes required to deviate from the agreed protocol because of unpredictable events. In addition, it was difficult to replicate the same exact conditions because of the human factors involved and the lack of control over some of the variables. Workshops in the classroom were often limited in time, involved a variable number of students and took place at different times of the day.

Despite the challenges, I was able to gather enough evidence to complete the design work. The experience matured was helpful in extracting valuable know-how from noisy data sets.

6.5.3 External Validity

External validity is concerned with the extrapolation of particular research findings beyond the immediate form of inquiry to the general.

While quantitative research, for example, using surveys aims at statistical generalisation and synthesis as methods to pursue external validity, case studies rely on analytic generalisation, whereby particular findings are generalised to some broader theory. The focus lies on an understanding and exploration of constructs, that is, usually the comparison of initially identified and/or developed theoretical constructs and the empirical results of single or multiple case studies (Riege, 2003).

In order to increase the external validity, several techniques were employed. The logic of the case study was replicated across different domains (Eisenhardt, 1989; Parkhe, 1993): most of the workshops involved students, but the tools were evaluated also with other target users (e.g. in P4), including researchers, employees from the municipality, entrepreneurs, programmers and freelancers. The domain of the study included in P4 is also unique since it was the only study to target IoT application for reflective learning.

The boundaries and scope of the research were defined in the research design phase (Marshall and Rossman, 2014). The outcome for each of the phases covered by the toolkits was clearly defined, as well as the target users of the toolkits and their attributes (see Chapter 4 for details).

Lastly, comparison of evidence with the extant literature in the domain of interest helped in clearly outlining the contributions and in generalising, always within the scope and boundaries of the research (Yin, 2017).

6.5.4 Reliability

Reliability refers to the demonstration that the operations and procedures of the research inquiry can be repeated by other researchers who then achieve similar findings; that is, the extent of findings can be replicated assuming that, for example, the interviewing techniques and procedures remain consistent (Riege, 2003).

In case study research, this can raise problems as people are not as static as measurements used in quantitative research, and even if researchers were concerned about ensuring that others can precisely follow each step, the results may still differ. Indeed, data on real-life events, which were collected by different researchers, may not converge into one consistent picture. However, possible differences also can provide a valuable additional source of information about the cases investigated (Riege, 2003).

The techniques used to increase reliability included the recording of observations and actions as concretely as possible (LeCompte and Goetz, 1982), the use of pilot studies to develop and refine the case study protocol (Eisenhardt, 1989; Vincent Mitchell, 1993; Yin, 2017), the use of multiple researchers who continually communicate about methodological decisions (LeCompte and Goetz, 1982), the mechanical recording of data (Nair and Riege, 1995), the development of a case study database to organise and document the mass of collected data (Lincoln and Guba, 1985) and finally the use of peer review and examination (LeCompte and Goetz, 1982). These techniques were applied for example by conducting pilot studies prior to the studies included in P2 and P4, by refining the methods in a collaborative manner within our research group, by systematically recording data through pictures of the artefacts and questionnaires, by redacting a spreadsheet containing all the details of each study, including version and type of the tools tested, details about the participants, type of data collected and research objectives.

Repeatability can be demonstrated by the existence of external publications employing the Tiles ideation toolkit in contexts similar to the ones presented in the papers included in this thesis (see Section 9.3 for details). In some cases, the ideation process directing the use of the Tiles cards was modified or extended (e.g. removing the cardboard). The results reported, however, are aligned with the findings presented in this thesis.

Relative to repeatability, limitations might affect the prototyping phase. This phase, in fact, was evaluated and tested only internally during the last months of my PhD. External adoption and independent evaluations of the prototyping toolkit are not yet possible because of the early stage of development of the hardware and software stacks.

6.6 Ethical Considerations

Data captured was handled in observance of the Norwegian University of Science and Technology (NTNU) policies. Occasionally younger users received a gift card after participating. In some workshops, the prize was awarded only to the best ideas generated, voted by the participants themselves. All the users involved in the studies signed a consent form which explained how the collected data was used. Younger users were required to have the consent form signed by a parent or a tutor. All the data collected was anonymised, kept confidential and not disclosed to third parties. Data contained in the articles thus do not include any information that can lead to the identification of a specific data subject. All the users were given the opportunity to withdraw their consent at any time and the data was destroyed after being analysed. While deciding which type of data to collect, we followed a *privacy by design* approach (Cavoukian et al., 2009), considering the data collected as a liability and not as an asset. In addition, compliance of data collection and handling with national legislation was assured by enforcing the regulations mandated by the Norwegian Centre for Research Data (NSD[2]).

[2] https://nsd.no/

6.7 User Centred Design Approach

In order to create more meaningful solutions, which are better tailored to the wishes and needs of the final users, designers have become more concerned about the stakeholders of their creations (Sanders and Stappers, 2008). In the landscape of human-centred design, UCD is one of the approaches that can help in that direction. UCD and co-design involve value creation in ongoing, productive collaboration with all relevant parties, with end-users playing a central role (Jansen and Maarten, 2017).

Benyon (2014) distinguished four ways in which UCD pays off:

- With close user involvement, products are more likely to meet the expectations of the users and their requirements. This leads to increased sales and lower costs incurred by customer services.
- System designers tailor products for people in specific contexts and with specific tasks, thereby reducing the chances of situations with a high risk of human error arising. UCD leads to safer products.
- Putting designers in close contact with users means a deeper sense of empathy emerges. This is essential in creating ethical designs that respect privacy and quality of life.
- By focusing on all users of a product, designers can recognise the diversity of cultures and human values through UCD, a step in the right direction towards creating sustainable businesses.

UCD is a design approach that foresees the active involvement of the stakeholders during the design process. Users play the role of *experts of their experience* and participate in knowledge and idea development (Sanders and Stappers, 2008). Research suggests that more innovative solutions can be obtained when co-design techniques are adopted (Trischler et al., 2018), while end-users are believed to have a better perspective of the problem at stake compared to designers, who are not necessarily familiar with the domain addressed as they are with the design methods used to tackle the problem.

Lockton et al. (2014), advocated the importance of understanding people. Their contexts and social practices of living and working are seen as fundamental to frame problems appropriately, namely, in a way that corresponds to the real lives of the people, instead of basing it on assumptions. UCD is an iterative process that aims to understand users and their contexts in all stages of design and development. UCD is pertinent to my work especially in connection to the toolkits produced during my PhD and described in Chapter 4. Multiple iterations were performed on the prototypes of the toolkits, addressing usability issues, extending the functionalities and improving the design. These iterations followed a UCD approach: field evaluations, direct user feedback and observations during field work informed each iteration on the design of the toolkit.

7 Results

This chapter summarises the papers that document the conducted research.

7.1 Overview of the Research Papers

Research work was published in peer-reviewed journals and conference proceedings. Seven of these publications are included in this thesis; three journal papers and four conference papers. The articles are summarised in this section, including the following:

- Title
- Authors and their roles in the paper
- Abstract of the paper
- Publisher
- A short description of how the paper relates to the research questions

Papers are reprinted in full in Part II of the thesis.

7.2 Paper 1

Title: Technology-Enhanced Smart City Learning: A Systematic Mapping of the Literature.

Authors: Francesco Gianni and Monica Divitini.

Contributions of the authors: Gianni led the research and the paper writing. He was actively involved in programming the study and collecting the data. The screening of the papers was performed mainly by Gianni, while articles coding was performed in equal measure by Gianni and Divitini. Divitini also provided general supervision for the research and the paper writing.

> *Abstract:* Smart cities are a popular and recognised research topic. In urban spaces, the learning factor is an important component for citizens and local communities. This paper presents a systematic mapping of the literature on smart city learning, with focus on how technology is used to

enhance smart city learning. The goal is to map the state of the art and to identify gaps in current research that can prompt new research in this area. Articles were collected from various online databases and relevant journal publications, selected according to defined inclusion/exclusion criteria. Abstracts were coded based on a number of criteria, including e.g. learning goal, used technology, and theoretical approach. Following the coding process results were analysed to identify themes. In the paper we shed light on the current understanding of smart city learning by (i) identifying common scenarios and learning settings; (ii) publication patterns; (iii) technical features in the supporting technology; (iv) learning theories and approaches that are mostly used; and (v) adopted type of research and research methods. The mapping shows that the concept of smart city learning is growing in popularity, with increasing number of publications in this area in the last years. However, the field is rather fragmented, with very different understanding of the concept. Smart city learning is also emerging as a very complex form of learning, with different stakeholders, learning activities, and technological solutions combined in rich eco-systems. The mapping also points out two largely unexplored areas of technological support, namely the Internet of Things (IoT) and the use of city-related data.

Published in: Interaction Design and Architecture(s) Journal - IxD&A 27:28–43, (2016).

Description: This paper maps the literature in the smart city learning domain. It provides grounding and identifies gaps in the literature, thus supporting the solutions designed in the subsequent works. The findings shed light on a field where modern technologies are seldom employed, and studies do not involve an heterogeneous sample of the citizens, nor they promote active participation. The paper started the investigation of RQ2, addressing how modern technologies can be tailored to the smart city learning domain.

7.3 Paper 2

Title: Tiles: A Card-Based Ideation Toolkit for the Internet of Things.

Authors: Simone Mora, Francesco Gianni and Monica Divitini.

Contributions of the authors: Mora created the toolkit and led the paper writing. Gianni created domain specific personas and scenarios used during the evaluation, analysed the data and wrote the second half of the paper. All the authors participated in the data collection during the evaluation workshops. Divitini provided general supervision for the research and the paper writing.

Abstract: The Internet of Things (IoT) offers new opportunities to invent technology-augmented things that are more useful, efficient or playful than their ordinary selves, yet only a few tools currently support ideation for the IoT. In this paper we present Tiles Cards, a set of 110 design cards and a workshop technique to involve non-experts in quick idea generation for augmented objects. Our tool aims to support exploring combinations of user interface metaphors, digital services, and physical objects. Then it supports creative thinking through provocative design goals inspired by human values and desires. Finally, it provides critical lenses through which analyse and judge design outcomes. We evaluated our tool in nine ideation workshops with a total of 32 participants. Results show that the tool was useful in informing and guiding idea generation and was perceived as appealing and fun. Drawing on observations and participant feedback, we reflect on the strengths and limitations of this tool.

Published in: Proceedings of the 2017 Conference on Designing Interactive Systems. DIS 2017. 587–598. Edinburgh, United Kingdom: ACM, (2017).

Description: This paper introduces the Tiles ideation toolkit, a card based design and ideation toolkit for IoT, see Fig. 4.2 for an overview on how it connects with the Tiles IoT framework. The paper discusses the very first design iteration on the toolkit, its evaluation, strengths and limitations emerged. The Tiles ideation toolkit represents the first step towards improving participation and co-design (Chapter 5) in the smart city learning domain. The paper investigates RQ1 and RQ2, addressing how non-experts can be included in the development process of technological applications for the smart city domain.

7.4 Paper 3

Title: Designing IoT Applications for Smart Cities: Extending the Tiles Ideation Toolkit.

Authors: Francesco Gianni and Monica Divitini.

Contributions of the authors: Gianni designed and created the extension of the toolkit, new cards were designed and produced from scratch, several new groups of cards were created, existing cards were redesigned, a new redesigned cardboard was created and a new improved design process (playbook) was devised. Gianni also supervised the user study, collected and analysed the data and led the paper writing. Divitini provided general supervision for the design of the extension, research and the paper writing.

Abstract: The internet of things (IoT) is gaining momentum as a technical tool and solution for a diverse range of societal challenges. These challenges include smart cities sustainability issues which are widely recognised by decision makers and societies. Despite this, few works try to tackle these challenges empowering citizens through IoT technologies. In this paper we describe how the Tiles toolkit, a card based idea generation toolkit for IoT, has been extended to support non-experts in creating ideas addressing societal challenges that affect modern smart cities. We briefly introduce the Tiles generic toolkit, then we describe in detail the extensions proposed on the cards, cardboard and how the new components are employed in a refined workshop protocol. We report the results obtained during a field study of the extended toolkit, where several groups of students collaborated to generate ideas involving IoT in the smart city. We discuss success and failures, drawing our conclusions after analysing quantitative and qualitative data collected during the workshop. We conclude the article reporting the lessons learned, critical considerations about our experience evaluating the extended toolkit and reflections on possible improvements for future works.

Published in: Interaction Design and Architecture(s) Journal - IxD&A 35:100–116, (2018).

Description: This paper presents the second and final design iteration on the Tiles ideation toolkit. New cards and a new cardboard are introduced, the workshop protocol is refined to improve usability and support for unattended activities without facilitators. Compared to P2, the smart city learning domain is better integrated in the toolkit and in the workshop protocol. With P3, the research work covering the first three phases of the Tiles IoT framework process (Fig. 4.1) is considered complete. The artefacts produced and the idea generation process involving non-experts were evaluated, leading to satisfactory results in terms of outcome of creative ideas, learning and support provided. The paper investigates RQ1 and RQ2, addressing how non-experts can generate ideas of IoT applications for smart cities, identifying a specific design strategy centred around tangible interfaces and augmented objects (Chapter 5), and finally linking to the subsequent rapid prototyping phase.

7.5 Paper 4

Title: Tiles-Reflection: Designing for Reflective Learning and Change Behaviour in the Smart City.

Authors: Francesco Gianni, Lisa Klecha and Monica Divitini.

Contributions of the authors: Gianni led the paper writing, Klecha designed the study and performed the theoretical grounding. Gianni and Klecha defined the extension of the toolkit, participated in the user studies, collected and analysed the data. Gianni performed the last iteration of the study, including the design and production of the cards for reflective learning and extended persona canvas. Gianni performed the user study, data collection and analysis. Divitini provided general supervision for the design of the extension, research and the paper writing.

Abstract: Modern cities are increasing in geographical size, population and number. While this development ascribes cities an important

function, it also entails various challenges. Efficient urban mobility, energy saving, waste reduction and increased citizen participation in public life are some of the pressing challenges recognised by the United Nations. Retaining liveable cities necessitates a change in behaviour in the citizens, promoting sustainability and seeking an increase in the quality of life. Technology possesses the capabilities of mediating behaviour change. A review of existing works highlighted a rather unilateral utilisation of technology, mostly consisting of mobile devices, employment of persuasive strategies for guiding behaviour change, and late end-user involvement in the design of the application, primarily for testing purposes. These findings leave the door open to unexplored research approaches, including opportunities stemming from the Internet of Things, reflective learning as behaviour change strategy, and active involvement of end-users in the design and development process. We present Tiles-Reflection, an extension of the Tiles toolkit, a card-based ideation toolkit for the Internet of Things. The extension comprises components for reflective learning, allowing thus non-expert end-users to co-create behaviour change applications. The results of the evaluation suggest that the tool was perceived as useful by participants, fostering reflection on different aspects connected to societal challenges in the smart city. Furthermore, application ideas developed by the users successfully implemented the reflective learning model adopted.

Published in: The Interplay of Data, Technology, Place and People for Smart Learning. SLERD 2018. *Smart Innovation, Systems and Technologies* 95:70–82. Aalborg, Denmark: Springer, Cham, (2019).

Description: This paper presents an extension of the Tiles ideation toolkit which targets the brainstorming and design of IoT applications for behaviour change. Smart cities are used as domain, and CSRL in employed as behaviour change strategy. New cards and a new storyboard are used to guide the users in the ideation process of IoT application which support reflective learning. This work demonstrates how the Tiles ideation toolkit can be extended to support specific domain problems and learning strategies. The paper investigates RQ2 by introducing a different design goal compared to P3 and P2, namely behaviour change through reflective learning.

7.6 Paper 5

Title: Designing IoT Applications in Lower Secondary Schools.

Authors: Anna Mavroudi, Monica Divitini, Francesco Gianni, Simone Mora and Dag R. Kvittem.

Contributions of the authors: Mavroudi led the paper writing and the theoretical learning framing. Divitini provided general supervision for the design of the toolkit, research, and contributed in writing the paper. Gianni and Mora designed the study, the tools and performed the data collection and analysis. Kvittem provided facilitation and guidance during the user study. All the authors were present on-site during the user study.

Abstract: The paper reports on a case study where four groups of lower secondary school students participated in a workshop and undertook the demanding role of designers of Internet of Things applications. In doing that, they made use of a dedicated inventor toolkit, which facilitated students' creative solutions to problems that can appear in the context of a smart city. From a pedagogical point of view, the workshop format is inline with the experiential learning approach. The paper presents a holistic student assessment methodology for this nice domain. In particular, to analyse the impact of the workshop for the students we used four different approaches: artefacts analysis of students' design solutions, classroom observations, a post-test and a survey. The results indicate that the intervention has promoted an effective teaching methodology for the basic conceptual and design aspects of the IoT for these lower secondary school students, but it has not addressed equally effectively the attitude-related aspects. Nonetheless, the participant students perceived the intervention as very satisfactory in terms of the IoT concept knowledge, smart cities learning, and problem-solving skills acquired, as well as in terms of enjoyment. The paper concludes on the learning gains of the intervention and discusses the motivation aspect for the teacher as well as for the students in this highly innovative learning experience.

Published in: Proceedings of IEEE Global Engineering Education Conference. EDUCON 2018. 1120–1126. Tenerife, Spain: IEEE, (2018).

Description: This paper evaluates the learning potential of the Tiles ideation toolkit in a classroom setting. Learning about the application domain, solution space and design methods are fundamental steps when targeting creative activities involving non-experts. The toolkit demonstrated to be effective in conveying knowledge about IoT, design, and in promoting smart city learning. Quantitative and qualitative data collected suggested the possibility to include the workshop as an integrated activity in the curriculum of the school. The paper investigates RQ2, addressing how the toolkit can be employed for specific goals: learning about IoT, smart cities and design methods.

7.7 Paper 6

Title: RapIoT Toolkit: Rapid Prototyping of Collaborative Internet of Things Applications.

Authors: Francesco Gianni, Simone Mora and Monica Divitini.

Contributions of the authors: Gianni led the paper writing, designed and supervised the development of the mobile application, designed and ran the evaluation study, collected and analysed the data and contributed in the development of the cloud based software and the electronic devices firmware. Mora contributed in writing the paper, designed the original architecture of the system, designed and produced the electronic devices. Both Gianni and Mora supervised the implementation of the system, and contributed in form of programming and testing. Divitini provided general supervision for research and paper writing.

Abstract: The Internet of Things holds huge promise in enhancing collaboration in multiple application domains. Bringing internet connectivity to everyday objects and environments promotes ubiquitous access to information and integration with third-party systems. Further, connected ‘things’ can be used as physical interfaces to enable users to cooperate, leveraging multiple devices via parallel and distributed actions. Yet creating prototypes of IoT systems is a complex task for developers non-expert in IoT, as it requires dealing with multi-layered hardware and software infrastructures. We introduce RapIoT, a software toolkit that facilitates the prototyping of IoT systems by providing an integrated set of technologies. Our solution abstracts low-level details and communication protocols, allowing developers non-expert in IoT to focus on application logic, facilitating rapid prototyping. RapIoT supports the development of collaborative applications by enabling the definition of high-level data type primitives and allowing interactions spread among multiple smart objects. RapIoT primitives act as a loosely coupled interface between generic IoT devices and applications, simplifying the development of systems that make use of an ecology of devices distributed to multiple users and environments. We illustrate the potential of our toolkit by presenting the development process of an IoT application ideated during a workshop with non-expert developers and addressing real-world challenges affecting smart cities. We conclude by discussing the strength and limitations of our platform, highlighting further possible uses for collaborative applications.

Published in: Journal of Future Generation Computer Systems. Elsevier, (2018).

Description: This paper presents RapIoT, a toolkit to support rapid prototyping of IoT applications, see Fig. 4.2 for an overview on how it connects with the Tiles IoT framework. The toolkit is composed by software to be deployed on the cloud, on mobile devices and on microcontrollers contained in embedded devices. It is designed having non-experts in mind, thus avoiding intricate deployment procedures and complex programming strategies. An event-based messaging protocol connects the electronic devices deployed on the field with the application logic running in the cloud. These messages are human readable and present a simple structure, thus allowing debugging and extensions with limited effort. The paper investigates RQ1 and RQ3, addressing the prototyping phase of IoT applications and how technology can support it best when non-experts are involved.

7.8 Paper 7

Title: Rapid Prototyping Internet of Things Applications for Augmented Objects: The Tiles Toolkit Approach.

Authors: Francesco Gianni, Simone Mora and Monica Divitini.

Contributions of the authors: Gianni wrote the paper, designed and ran the evaluation study, collected and analysed the data, designed and produced one of the electronic devices and programmed both of them. Mora contributed in writing the paper, designing and producing the second electronic device, and in running the evaluation study. Divitini provided general supervision for research and paper writing.

Abstract: Designing and prototyping for IoT have historically required a diverse range of skills and a set of tools that individually supported only a fraction of the whole process, not being designed to work together. These tools usually require a certain level of proficiency in design methods, programming or electronics, depending on the phase addressed. Previous works on the Tiles Ideation toolkit and the RapIoT software framework demonstrated how the design phase can be democratised and how a simple programming paradigm can make coding for IoT a task accessible to non-experts. With this work we present and evaluate the process and the technologies involved in the programming and prototyping phase of an IoT application. The Tiles Square and the Tiles Temp are introduced, these two electronic devices complement and support IoT prototyping. They are designed to work in conjunction with the Tiles Ideation toolkit and are supported by the RapIoT software framework, allowing non-experts to augment and program everyday objects. We illustrate the potential of this approach by presenting the results obtained after workshops with 44 students. We conclude by discussing strengths and limitations of our approach, highlighting the lessons learned and possible improvements.

Published in: Ambient Intelligence. AmI 2018. *Lectures Notes in Computer Science.* 11249:204–220. Larnaca, Cyprus: Springer, Cham, (2018).

Description: This paper evaluates the rapid prototyping of IoT application ideas generated with the Tiles ideation toolkit. Two electronic devices were designed and built, allowing non-experts to augment objects by simply attaching them to things. The RapIoT toolkit is used to support the users in programming the application behaviour. RapIoT also provided network connectivity to the electronic devices attached to the objects, and assisted the users in the initial physical deployment procedure. The evaluation demonstrated how non-experts were able to complete the steps required to transform an application idea into a tangible prototype, embedding into an object the application behaviour envisioned. Such outcome also validates the transition between the idea generation and design phases, where the Tiles ideation toolkit is employed, and the prototyping phase. The paper contributes in answering all the research questions. It addresses mainly the prototyping phase (RQ3), but it is strongly tied to the preceding articles, building on the assumptions, boundaries and design choices that drove the entire development process.

8 Contributions

The contributions connected to the work presented in this thesis were divided into five areas:

1. Improved understanding of the holistic process supporting non-experts in idea generation, design and prototyping of IoT applications.
2. Contribution to the design, implementation and evaluation of a card-based design toolkit for idea generation, allowing non-experts to generate concepts of IoT applications.
3. Contribution to the design, implementation and evaluation of a rapid prototyping toolkit for tangible exploration of ideas based on smart augmented objects.
4. Improved understanding of educational methods to support learning about smart cities and IoT.
5. Knowledge in the definition of a technological framework to overcome recognised challenges and limitations affecting the IoT domain.

Contribution 1 covers an integrated framework to support non-experts during the complete user-journey from idea generation to prototyping of IoT applications. Contribution 2 includes the work connected to the creation of the Tiles toolkit for idea generation. Contribution 3 is related to the development of the RapIoT toolkit for rapid prototyping. Contribution 4 maps the tools created into the context of SCL education. Finally Contribution 5 proposes a strategy to further improve the tools created, based on recognised technical standards and improved understanding of the domain.

Table 8.1 summarises the contributions provided by the papers. In Fig. 8.1, a contribution map is presented, including the papers, the contributions and the research domains.

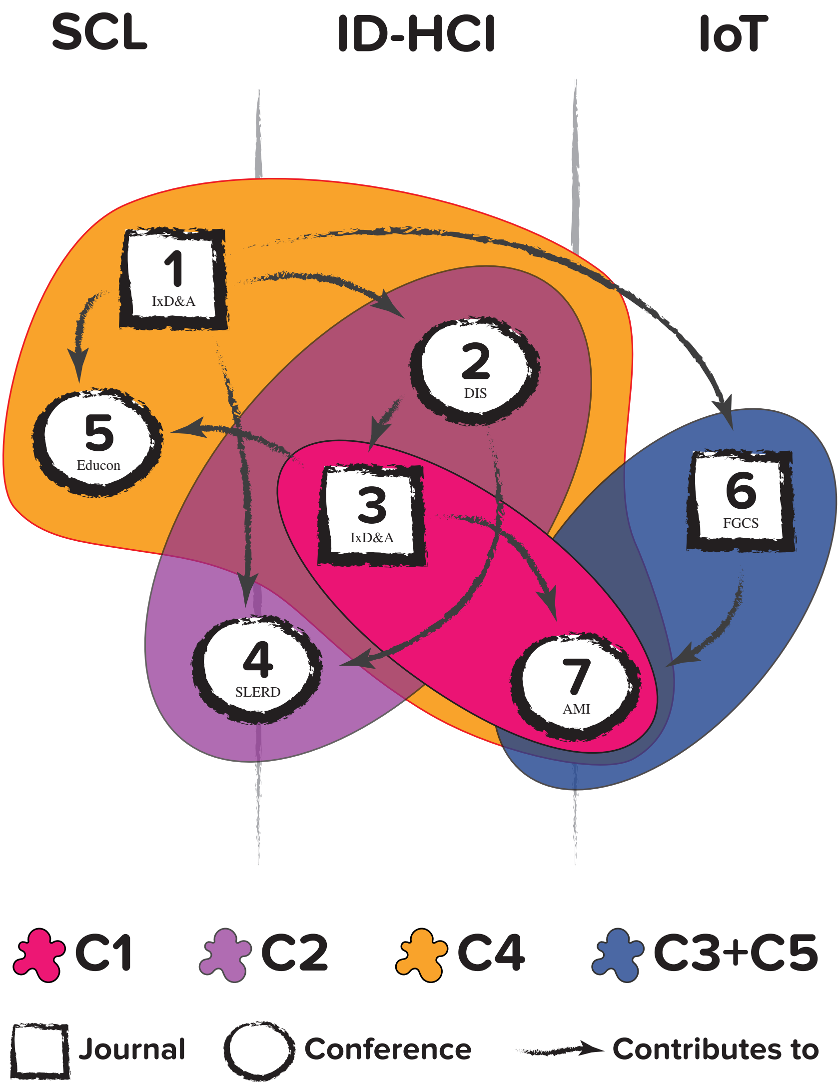


Figure 8.1: A schema of the contributions. On top, from left to right: the domains of SCL, ID, HCI and IoT.

Table 8.1: Mapping of contributions with papers.

	C1	C2	C3	C4	C5
Paper 1				•	
Paper 2		•		•	
Paper 3	•	•		•	
Paper 4		•			
Paper 5				•	
Paper 6			•		•
Paper 7	•		•	•	•

8.1 C1: Improved understanding of the holistic process supporting non-experts in idea generation, design and prototyping of IoT applications.

Contribution 1 is related to the theoretical definition, design and evaluation of an integrated framework supporting non-experts in idea generation and rapid prototyping of IoT applications based on augmented objects. In the context of this holistic process, the work presented in the thesis can be divided into two main blocks. The first one addresses the initial idea generation and design phases, where non-experts learn about the domain and generate an idea. The second block aims at creating a tangible prototype of the idea generated and includes the programming of the augmented object behaviour. The process is tuned to be integrated, reusing the same conceptual building blocks during the different phases. It allows gradually building on the knowledge acquired, steadily supporting non-experts without demanding professional skills in any of the steps required. The definition of input/output primitives (Section 4.3) is an example of these conceptual building blocks that facilitate the transition between the design and prototyping phases. I designed and developed the majority of the input/output primitives that were used in the studies included in P3, P6 and P7, both in terms of cards included in the Tiles ideation toolkit and software of the RapIoT toolkit. Special attention to the nature of this holistic process is reserved in P3 and P7, which also represent at best the work pertaining to each of the two blocks. For a detailed description of the complete framework, see Chapter 4. Part of this contribution are also the lessons learned and the guidelines formalised as a result of my experience on the field (see P3 §6 and P7 §5, §6).

This novel approach differs from the one adopted by most of the toolkits already available, which usually support either the design and brainstorming part or the prototyping and programming one, with no facilitation in bridging the two stages. A production-ready toolkit embracing this framework can be beneficial in education and design and for tangible exploration of creative ideas.

8.2 C2: Contribution to the design, implementation and evaluation of a card-based design toolkit for idea generation, allowing non-experts to generate concepts of IoT applications.

Contribution 2 of the thesis comprises new knowledge about how theoretical concepts of co-design and tangible user interaction (Chapter 5) can inform the design, implementation and evaluation of a design and idea generation toolkit for IoT. The toolkit was developed during several design iterations to support non-experts in idea generation and design of IoT applications; its initial version is described in P2. The toolkit was designed to be frictionless and easy to adopt for users without previous knowledge of IoT or design methods. Some of the improvements developed during the first design iteration and the specialisation towards the smart city domain are described in P3. At the end of each design iteration, the artefacts were empirically evaluated, and the feedback gathered informed the following version of prototypes. The final artefact produced by the work related to C2 was a new version of the generic Tiles ideation toolkit and additional specialisations addressing IoT applications for reflective learning and smart cities. More in particular, I redesigned the cardboard (Fig. 4.4), modified some of the already existing cards, designed and added new cards and card categories (P3 §3) and added an extension to the storyboard and personas canvas to support IoT applications for reflective learning (P4 §3). I used the insights gathered on the field to prepare a set of instructions to guide the users during group-based brainstorming and design activities. Starting from these instructions, I refined the *playbook* reported on the cardboard, initially described in P2. The new *playbook* was then successfully employed in the version of the toolkit used during P3, P4, P5 and P7. I developed and refined such guidelines to provide a toolkit that is self-supported and does not need direct supervision by experts to be used. In order to facilitate adoption, especially in educational contexts, I designed the process to be self-contained, short-lasting from start to finish and entertaining. The toolkit, the idea generation process and all the materials were open source and were made available for free under a creative commons licence.

The Tiles ideation toolkit supports co-design as a strategy to involve different stakeholders in the brainstorming phase of an IoT application. The group-based activity, the use of specific cards designed to spark creativity and reflection and the physicality of combining artefacts to define the seed of the idea all helped in provide a shared understanding of the problem and solution strategy, despite the diversities in skills and backgrounds of the participants. The toolkit is a viable instrument also for co-design workshops. Depending on the goal of the design workshop, a facilitator or expert might be necessary to support the users. This scenario emerged during the studies described in P4, where I specialised the toolkit towards IoT applications for reflective learning. Given the complexity of this approach for non-experts, a facilitator was needed to support the users during the workshops.

Contribution 2 can be a resource for researchers, designers, students and any user falling in the category of non-expert, who is interested in learning about IoT and generating ideas of IoT applications based on augmented objects and tangible interfaces. Although the toolkit was born to be generic, I subsequently specialised it for the smart city domain and for IoT application for reflective learning. This demonstrates its flexibility and the opportunity for it to be repurposed for other domains with limited effort.

8.3 C3: Contribution to the design, implementation and evaluation of a rapid prototyping toolkit for tangible exploration of ideas based on smart augmented objects.

The third contribution of the thesis includes the design, implementation and evaluation of an electronic and programming toolkit to prototype IoT ideas of augmented objects. The final prototypes produced are examples of tangible interfaces (Chapter 5), and the toolkit itself is oriented into supporting rapid and tangible exploration of idea concepts. The RapIoT toolkit is designed to connect and build on the outcome of the brainstorming and design phases, covered by C2. The RapIoT toolkit is composed of several hardware devices and software layers deployed in a cloud application, a development environment, a mobile app and a firmware for microcontrollers for embedded devices. For a more detailed description, see Chapter 4. The software structure of the toolkit is mainly described in P6, whereas a more extensive evaluation where non-experts created prototypes of augmented objects using the toolkit is reported in P7. In order to allow non-experts to create a smart object and program its behaviour, the toolkit aimed to keep the entry barriers low, as in the preceding phases. The toolkit includes electronic *stickers* that can be attached to regular objects to provide sensing and actuation capabilities. Since several *stickers* can be used in a single application, it is possible to prototype applications supporting distributed input and output, spread into several interconnected objects. I designed, produced and tested one of the two electronic *stickers*, during two design iterations. Ease of use and rapid deployment are guaranteed by the absence of any wire and the use of a simplified language for programming and by not requiring the installation of any software. I contributed in writing part of the cloud software layer of RapIoT, I wrote from scratch the firmware of one of the *stickers* and rewrote and extended the firmware of the second one. I supervised and directed the development of the mobile application, which was assigned as a project to a group of students.

Contribution 3 can provide valuable insights into how to design a toolkit for tangible exploration of ideas for non-experts. The RapIoT toolkit itself is released under a creative commons licence and is free to use. It can act as a resource for non-experts, designers, students and makers. Crafting a physical prototype is an enriching experience that conveys knowledge, builds experience and provides a tangible artefact to validate use cases and idea concepts. Artefacts can also represent a shared means to further develop the discussion and collaboration around the original idea.

8.4 C4: Improved understanding of educational methods to support learning about smart cities and IoT.

The fourth contribution of the thesis is related to the educational outcome, mainly connected to the use of the Tiles ideation toolkit during school hours, in the context of SCL. The user studies described in most of the papers included in this thesis had a twofold purpose: (i) to evaluate a specific iteration of the toolkit design and (ii) to support the learning process of the students involved in the evaluation. These user studies involved students aged from 13 to 27. The ideation workshop was included in the programme of courses on IoT or design or performed as a standalone activity during school hours. While collecting data on the aspects related to the toolkit, we were also able to extend the investigation towards the learning outcome of the idea generation and prototyping activities supported by the toolkits. We recorded good feedback in terms of increased knowledge about the challenges connected to smart cities, the definition of IoT and its design space and increased interest in programming and tinkering.

Contribution 4 provides guidelines and an overview of the lessons learned while deploying the toolkits for educational purposes. I assessed these two aspects in P7 §5 to §7, while a more elaborated evaluation protocol for the learning outcomes of the workshop was devised in P5. During these studies, knowledge was gathered in adapting the creative activity to students of different ages and with diverse backgrounds.

8.5 C5: Knowledge in the definition of a technological framework to overcome recognised challenges and limitations affecting the IoT domain.

The fifth contribution condenses the experience gathered in working with IoT design and prototyping toolkits and IoT systems in general. During the research activities described in this thesis, I put considerable effort in investigating and experimenting with different IoT technologies, software frameworks, sensors, microcontrollers and networking solutions. Some of the outlined challenges affecting modern IoT solutions were identified, and the vision for an improved IoT technical framework was drafted, inspired and supported by emerging movements, standards and publications targeting the same shortcoming. Some of these challenges were broad and generic. However, whenever possible, the IoT model I envisioned and described in Chapter 4 was specialised to address the requirements dictated by having non-experts as target users.

Contribution 5 provides the description of an IoT technical framework to support rapid prototyping of smart objects, having non-experts as target users. The envisioned solutions aim to be open, accessible and future-proof, following the philosophy adopted by the web. See Chapter 4 for a more detailed description of the framework.

9 Evaluation

This chapter describes the evaluation of the contributions presented in Chapter 8 in connection with the research questions. In addition, limitations are discussed, and the external impact of the research work is summarised.

9.1 Evaluation of Research Questions

MRQ: How can non-expert users be supported during brainstorming, design and tangible exploration of ideas of IoT applications for SCL?

The main research question is answered by Contributions 1, 2 and 3. Using co-design workshops, it was demonstrated how concepts from theories in tangible user interaction can be deployed into a suite of tools, applications, methods and artefacts. This integrated suite was demonstrated to be useful and effective in supporting non-experts in generating ideas and prototypes of tangible interfaces and smart augmented objects. Different tools addressed different phases of the process, whereas a holistic framework was designed to support the users throughout the whole journey.

RQ1: How can the phases of brainstorming, design and tangible exploration of IoT ideas be connected and integrated to provide an experience adapted to the skills of non-expert users?

The answer to the first research question is provided by Contribution 1. The work included in C1 describes in detail the tools and co-design methods employed during the phases included in the process defined by the Tiles IoT framework. Emphasis is placed on how the transition between the idea and the prototype is facilitated, maintaining a complexity level suitable for non-experts.

RQ2: Which kind of brainstorming and design tools can be employed to generate IoT application ideas? And how can they be specialised to target specific goals and application domains?

This research question is answered by Contributions 2 and 4. C2 includes the studies where the Tiles ideation toolkit was firstly introduced and the ones where it was specialised in the domains of smart cities and IoT applications for reflective learning. C4 explores how the toolkits and the methods developed can be employed to support SCL.

RQ3: Which IoT technologies and architectures can efficiently support prototyping and tangible exploration of ideas for non-expert users?

This research question is answered by Contribution 3 and 5. A technical architecture to support prototyping of IoT applications is described in C5. Its implementation into a rapid prototyping toolkit is then proposed and evaluated during prototyping workshops with non-expert users. C3 focuses on these phases of programming, prototyping and tangible exploration of ideas.

9.2 Limitations of the Research Approach

The limitations of the research approach can be divided into factors connected to the nature of the case studies performed and limitations related to the role of the learning methods mentioned in the thesis. Although it is important to clarify the boundaries of my research, the limitations here described are mostly related to the time frame and resources at my disposal during the PhD period. Such limitations however did not significantly affect the scope and validity of the results described in Chapters 7 and 8.

9.2.1 Case Studies

The limited time at our disposal during some of the case studies often prevented to perform personal interviews with the users. We were able to perform interviews during the first half of the studies, however when the evaluation workshops started to involve larger groups of users, limited time and resources prevented any case-by-case in depth assessment with the users.

The diverse conditions in which the studies took place affected users and researchers. When generalising the findings cross-case, conditions for each study had to be taken into account in order to provide a robust interpretation of the phenomenon. This heterogeneity added complexity and limited the possibilities of generalisation of the results.

Long-term effects were difficult to assess for the same reasons described above. Especially for the prototyping phase, it was not possible to assess more than a short prototyping and programming iteration, while long-term dynamics connected to an iterative prototyping process might have led to additional understandings.

Given the different conditions in which the workshops took place and the human factors involved, it was challenging to replicate the same protocol in all the studies. Human factors also led to unforeseen situations that had to be addressed on site, quickly and case-by-case.

9.2.2 Role of Learning

Learning and related practices are recalled and addressed in several forms within the work described in this thesis. For example (i) learning is part of SCL, the application domain used during the workshops, (ii) the learning outcome of the users in terms of acquired knowledge about IoT and other topics associated with the workshop activities were evaluated during some of the studies, (iii) an extension of the Tiles ideation toolkit for IoT applications for reflective learning was created and (iv) a group of cards included in the Tiles ideation toolkit encouraged the users to reflect on their idea by analysing it under a given perspective.

The toolkits developed were not originally designed as learning tools, but their potential in facilitating learning emerged during the evaluation studies. For this reason, I decided together with my research group to include an assessment of the learning outcome in the evaluation. However, learning is not part of my research questions, in fact assessing the learning outcome was not the main objective of the case studies performed.

Learning as part of SCL was not evaluated since SCL was used mainly as an application domain, to provide the users a design space for their IoT ideas. The ultimate goal of the studies was not to create an IoT product to solve the challenges connected to SCL, since this would have required an evaluation of the fully developed ideas deployed on field, a task out of the scope of my PhD.

The extension of the Tiles ideation toolkit covering IoT applications for reflective learning (P4) contributed to demonstrate how the toolkit can be specialised to target specific goals and application domains (RQ2). For the same reasons described above, the efficacy of the ideas developed by the participants during the workshops was not tested on the field, however we evaluated if the necessary characteristics of an application for reflective learning were included in the concepts.

Finally, the *criteria* cards of the Tiles ideation toolkit invite the users to reflect and adapt their idea during the design process, proposing some reflection lenses. However, this design phase cannot be considered as a structured reflective learning process, given that for example it could simply consist in adapting the idea to increase its market potential or feasibility.

9.3 Exploitation and External Impact of Research

The Tiles ideation toolkit has been largely and independently employed in the research, governance, industry and education sectors from institutions in Europe, America, Asia and Australia. A few examples are the works of Gennari et al. (2017), Sintoris at al. (2018), Avouris et al. (2018) and Zhai et al. (2018). The Tiles ideation toolkit was also selected by an independent group of researchers to support an ideation and rapid prototyping workshop during the CHI Conference on Human Factors in Computing Systems, in 2018 (Angelini et al., 2018).

The Tiles ideation toolkit was employed at NTNU during workshops as part of the master-level university courses in *Cooperation Technology and Social Media* (TDT4245) at the Department of Computer Science, *Prototyping Interactive Media* (TPD4126) at the Department of Design and *Design of Communicating Systems* (TTM4115) at the Department of Information Security and Communication Technology.

A small community formed around the Tiles project, with particular emphasis on the Tiles ideation toolkit, but with high interest also in the tools for programming and rapid prototyping. Around 10 groups approached us directly to ask for guidance, customisation and assistance in running the workshop. Thanks to the open-source licence of the cards, cardboard and workshop technique, several contributed to creating extensions for different domains, like smart buildings, or creating typesets to ease printing and collaboration tasks.

A company was started to refine and integrate the prototypes produced during the research activities, with the objective of developing a kit suitable for distribution.

10 Conclusions and Future Work

The work described in this thesis focused on enabling non-experts to generate ideas and prototypes of IoT applications involving augmented objects and tangible interfaces.

The main research method adopted was design science research. Several user studies were performed to evaluate the tools developed and the creative process supporting the users from idea generation to prototyping. The work was grounded in the literature about SCL, co-design, tangible interfaces and user-centred design. Prototypes and tools were produced in the form of a design toolkit, electronic devices for object augmentation, mobile applications and a software toolkit. Each of them was developed through multiple design iterations, producing various intermediate versions at different levels of fidelity. The scientific work has been published in peer-reviewed journals and conference proceedings, and seven of these publications were included in this thesis.

The research questions were answered by five contributions, hereafter summarised in a set of conclusions that also delineate future work.

Conclusion 1

The experience matured and the tools produced can be used to support non-experts in learning about IoT and smart cities, generating an IoT application idea starting from a real problem and prototyping such idea into a programmable demonstrator. This emerged as a complex process to define and evaluate, involving a diverse set of skills that require team effort in the areas of learning, social sciences, programming, networking protocols, electronics and embedded hardware and software. Prototypes covering each of these areas were created and successfully evaluated for their impact on supporting non-experts throughout the whole creative process (P3, P7). Lessons learned and feedback gathered drove the definition of new theories and guidelines (P7).

Additional work is required to finalise the design of the tools, especially the ones covering the prototyping phase. Given the positive feedback collected, future work will generalise tools and methods to cover other application domains in addition to smart cities.

Conclusion 2

The holistic IoT ideation process, spanning from brainstorming to prototyping, already covered in C1, was investigated in two phases. The first phases covered the process of idea generation and design, during which the users familiarised with the application domain and the solution space and produced an IoT application idea while brainstorming in group-based workshops. These activities were supported through the Tiles ideation toolkit, a card-based toolkit developed and evaluated in multiple iterations (P2, P3 and P4). We received excellent feedback during the user studies, and the toolkit was independently adopted by schools and institutions all over the world. We experimented with extensions targeting specific domains and learning strategies (P4).

Future work might include the internationalisation of the cards and the cardboard, in order to promote adoption in non-English-speaking countries. Minor enhancements in the card decks are also planned to further reduce the complexity and increase the ease of use. Finally, new cards could eventually be created to specialise the toolkit for specific application domains, like healthcare or smart homes.

Conclusion 3

The final phases of the process covered in C1 address rapid prototyping and tangible exploration of ideas. These phases were designed to integrate with the idea generation phase and to allow non-experts to produce a prototype of the augmented objects envisioned. The RapIoT toolkit (P6) supported the technical infrastructure employed by the users during programming and prototyping. The technologies include a cloud development environment, an application for mobile devices, a communication protocol and a firmware for embedded devices. The process and the technical solutions were successfully evaluated (P7) through user studies where non-experts developed and programmed prototypes of augmented objects.

Future work will be oriented towards revising the technical tools in order to adopt open standards, increase the efficiency and simplify the programming paradigm and the deployment procedure.

Conclusion 4

While running the field studies, the educational potential of the Tiles ideation toolkit naturally emerged. A side research line was then dedicated to exploring more systematically the outcome in terms of learning (P5). When the Tiles ideation toolkit was employed as a learning tool, we received good feedback in terms of improved knowledge about IoT and smart cities. The combination between technology and domain expertise was also beneficial in conveying SCL (P2, P5 and P7).

Future improvements to support the learning outcome can include better integration into the curricula of the schools. Internationalisation of the tools might also benefit the younger students, who occasionally find it difficult to understand the terminology employed without being assisted by the teacher.

Conclusion 5

In the process of designing, developing and evaluating the tools and methods described in this thesis, theoretical and technical knowledge about the IoT domain was acquired. Design and technical challenges became evident while experimenting and were later confirmed by reviewing the relevant domain literature (Chapter 4). Based on such literature, a descriptive draft of an IoT framework was redacted. This framework aims on the one hand to overcome the identified challenges and on the other hand to extend the support towards a concept of IoT that includes humans in the loop, standardising tangible interaction primitives and object manipulation. Future work will include a formal definition of the framework and an initial implementation that integrates with the architecture described in C3.

In conclusion, this thesis has developed knowledge and tools about a holistic process supporting non-experts in learning, ideation and prototyping of IoT applications for augmented objects. My investigation used SCL as a domain case, but the basis for generalising the theories and technologies developed has been settled. Commercial exploitation is currently being explored, and future work aims to improve the prototypes created and to explore new application domains while maintaining the theoretical foundations that backed the work presented here.

References

Alkar, Ali Ziya and Umit Buhur (2005). 'An Internet Based Wireless Home Automation System for Multifunctional Devices'. In: *IEEE Transactions on Consumer Electronics* 51.4, pp. 1169–1174.

Angelidou, Margarita (2014). 'Smart City Policies: A Spatial Approach'. In: *Cities* 41, S3–S11.

Angelini, Leonardo et al. (2018). 'Internet of Tangible Things: Workshop on Tangible Interaction with the Internet of Things'. In: *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems*, W14.

Ashton, Kevin (2009). 'That "Internet of Things" Thing'. In: *RFiD Journal* 22.7, pp. 97–114.

Avouris, Nikolaos, Irene Mavrommati and Christos Sintoris (2018). 'Designing through Ideation Cards for Internet of Things: Can Cards Help Engineers out of the Box?' In: *Proceedings of the 11th Panhellenic Conference with International Participation Information and Communication Technologies in Education.*

Baldwin, Carliss Young and Kim B Clark (2000). *Design Rules: The Power of Modularity.* Vol. 1. MIT press.

Batty, Michael et al. (2012). 'Smart Cities of the Future'. In: *The European Physical Journal Special Topics* 214.1, pp. 481–518.

Beigl, Michael, Hans-W Gellersen and Albrecht Schmidt (2001). 'Mediacups: Experience with Design and Use of Computer-Augmented Everyday Artefacts'. In: *Computer Networks* 35.4, pp. 401–409.

Benyon, David (2014). *Designing Interactive Systems: A Comprehensive Guide to HCI, UX and Interaction Design.* Pearson Edinburgh. 640 pp. ISBN: 1-4479-2011-2.

Bibri, Simon and John Krogstie (2016). 'Big Data Analytics and Context-Aware Computing for Smart Sustainable Cities of the Future'. In: *NOBIDS Symp.* Vol. 1818, pp. 4–17.

— (2017a). 'On the Social Shaping Dimensions of Smart Sustainable Cities: A Study in Science, Technology, and Society'. In: *Sustainable Cities and Society* 29, pp. 219–246.

— (2017b). 'Smart Sustainable Cities of the Future: An Extensive Interdisciplinary Literature Review'. In: *Sustainable Cities and Society* 31, pp. 183–212.

Booth, Tracey and Simone Stumpf (2013). 'End-User Experiences of Visual and Textual Programming Environments for Arduino'. In: *International Symposium on End User Development*, pp. 25–39.

Booth, Tracey, Simone Stumpf et al. (2016). 'Crossed Wires: Investigating the Problems of End-User Developers in a Physical Computing Task'. In: *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pp. 3485–3497.

Brown, Tim (2005). 'Strategy by Design'. In: *Fast Company*. URL: https://www.fastcompany.com/52795/strategy-design.

Cao, Jinwei et al. (2006). 'Interactions between System Evaluation and Theory Testing: A Demonstration of the Power of a Mulitfaceted Approach to Systems Research'. In: *Journal of Management Information Systems* 22.4, pp. 207–235.

Caragliu, Andrea, Chiara Del Bo and Peter Nijkamp (2011). 'Smart Cities in Europe'. In: *Journal of Urban Technology* 18.2, pp. 65–82. ISSN: 1063-0732. DOI: 10.1080/10630732.2011.601117.

Carneiro, Gabriela and Zhu Li (2011). 'I—o Cards: A Tool to Support Collaborative Design of Interactive Objects'. In: *Proceedings of DESIRE*, pp. 357–2.

Cavoukian, Ann et al. (2009). 'Privacy by Design: The 7 Foundational Principles'. In: *Information and Privacy Commissioner of Ontario, Canada* 5.

Chen, Yuang and Thomas Kunz (2016). 'Performance Evaluation of IoT Protocols under a Constrained Wireless Access Network'. In: *Selected Topics in Mobile & Wireless Networking (MoWNeT), 2016 International Conference On*, pp. 1–7.

Corsi, Marco (2002). 'The Child Friendly Cities Initiative in Italy'. In: *Environment and Urbanization* 14.2, pp. 169–179.

Cross, Nigel (1971). 'Design Participation'. In: *Proceedings of the Design Research Society's Conference*. Manchester: Academy Editions.

Cvijikj, Irena Pletikosa and Florian Michahelles (2011). 'The Toolkit Approach for End-User Participation in the Internet of Things'. In: *Architecting the Internet of Things*. Springer, pp. 65–96.

Cypher, Allen and Daniel Conrad Halbert (1993). *Watch What I Do: Programming by Demonstration*. MIT press.

DiSalvo, Carl, Phoebe Sengers and Hrönn Brynjarsdóttir (2010). 'Mapping the Landscape of Sustainable HCI'. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 1975–1984.

Dourish, Paul (2004). *Where the Action Is: The Foundations of Embodied Interaction*. MIT press.

Eisenhardt, Kathleen M (1989). 'Building Theories from Case Study Research'. In: *Academy of management review* 14.4, pp. 532–550.

Flick, Uwe (1992). 'Triangulation Revisited: Strategy of Validation or Alternative?' In: *Journal for the theory of social behaviour* 22.2, pp. 175–197.

Fu, Zhiyong and Xu Lin (2014). 'Building the Co-Design and Making Platform to Support Participatory Research and Development for Smart City'. In: *International Conference on Cross-Cultural Design*. Springer, pp. 609–620.

Galán-García, José L, Gabriel Aguilera-Venegas and Pedro Rodríguez-Cielos (2014). 'An Accelerated-Time Simulation for Traffic Flow in a Smart City'. In: *Journal of Computational and Applied Mathematics* 270, pp. 557–563.

Gennari, Rosella et al. (2017). 'Design of IoT Tangibles for Primary Schools: A Case Study'. In: *Proceedings of the 12th Biannual Conference on Italian SIGCHI Chapter*. CHItaly '17. Cagliari, Italy: ACM, 26:1–26:6. ISBN: 978-1-4503-5237-6. DOI: 10.1145/3125571.3125591.

Greenberg, Saul (2007). 'Toolkits and Interface Creativity'. In: *Multimedia Tools and Applications* 32.2, pp. 139–159.

Griggs, Steve (1987). 'Analysing Qualitative Data.' In: *Journal of the Market Research Society*.

Gubbi, Jayavardhana et al. (2013). 'Internet of Things (IoT): A Vision, Architectural Elements, and Future Directions'. In: *Future Generation Computer Systems* 29.7, pp. 1645–1660.

Guinard, Dominique and Vlad Trifa (2016). *Building the Web of Things: With Examples in Node.Js and Raspberry Pi*. Manning Publications Co.

Gulliksen, Jan et al. (2003). 'Key Principles for User-Centred Systems Design'. In: *Behaviour and Information Technology* 22.6, pp. 397–409.

Hernández-Muñoz, José M et al. (2011). 'Smart Cities at the Forefront of the Future Internet'. In: *The Future Internet Assembly*, pp. 447–462.

Hevner, Alan (2007). 'A Three Cycle View of Design Science Research'. In: *Scandinavian journal of information systems* 19.2, p. 4.

Hevner, Alan and Samir Chatterjee (2010). *Design Research in Information Systems - Theory and Practice*. Vol. 22. Integrated Series in Information Systems. Springer Science & Business Media.

Hirschman, Elizabeth (1986). 'Humanistic Inquiry in Marketing Research: Philosophy, Method, and Criteria'. In: *Journal of marketing Research*, pp. 237–249.

Hollands, Robert (2008). 'Will the Real Smart City Please Stand Up?' In: *City* 12.3, pp. 303–320. ISSN: 1360-4813.

Hornecker, Eva (2010). 'Creative Idea Exploration within the Structure of a Guiding Framework: The Card Brainstorming Game'. In: *Proceedings of the Fourth International Conference on Tangible, Embedded, and Embodied Interaction*. ACM, pp. 101–108.

Hornecker, Eva and Jacob Buur (2006). 'Getting a Grip on Tangible Interaction: A Framework on Physical Space and Social Interaction'. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 437–446.

Houben, Steven et al. (2016). 'Physikit: Data Engagement through Physical Ambient Visualizations in the Home'. In: *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM, pp. 1608–1619.

Innes, Judith and David Booher (2004). 'Reframing Public Participation: Strategies for the 21st Century'. In: *Planning theory & practice* 5.4, pp. 419–436.

Irvin, Renee and John Stansbury (2004). 'Citizen Participation in Decision Making: Is It Worth the Effort?' In: *Public administration review* 64.1, pp. 55–65.

Ishii, Hiroshi and Brygg Ullmer (1997). 'Tangible Bits: Towards Seamless Interfaces between People, Bits and Atoms'. In: *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, pp. 234–241.

Jacob, Robert et al. (2008). 'Reality-Based Interaction: A Framework for Post-WIMP Interfaces'. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 201–210.

Jansen, Stefanie and Pieters Maarten (2017). *The 7 Principles of Complete Co-Creation*. BIS Amsterdam. ISBN: 978-90-6369-473-9.
Kortuem, Gerd et al. (2010). 'Smart Objects as Building Blocks for the Internet of Things'. In: *IEEE Internet Computing* 14.1, pp. 44–51.
Kuniavsky, Mike (2010). *Smart Things: Ubiquitous Computing User Experience Design*. Elsevier.
Lansdown, Gerison (2010). 'The Realisation of Children's Participation Rights: Critical Reflections'. In: *A handbook of children and young people's participation: Perspectives from theory and practice*.
LeCompte, Margaret and Judith Preissle Goetz (1982). 'Problems of Reliability and Validity in Ethnographic Research'. In: *Review of educational research* 52.1, pp. 31–60.
Ledo, David et al. (2018). 'Evaluation Strategies for HCI Toolkit Research'. In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. CHI '18. Montreal QC, Canada: ACM, 36:1–36:17. ISBN: 978-1-4503-5620-6. DOI: 10.1145/3173574.3173610.
Lidwell, William, Kritina Holden and Jill Butler (2010). *Universal Principles of Design, Revised and Updated: 125 Ways to Enhance Usability, Influence Perception, Increase Appeal, Make Better Design Decisions, and Teach through Design*. Rockport Pub.
Lieberman, Henry et al. (2006). 'End-User Development: An Emerging Paradigm'. In: *End User Development*. Springer, pp. 1–8.
Lincoln, Yvonna and Egon Guba (1985). *Naturalistic Inquiry*. Vol. 75. Sage.
Lockton, Dan et al. (2014). 'Designing with People in Sustainability and Behaviour Change Research'. In: Design Research Society.
Lombardi, Patrizia et al. (2012). 'An Advanced Triple-Helix Network Model for Smart Cities Performance'. In: *Regional Development: Concepts, Methodologies, Tools, and Applications*. IGI Global, pp. 1548–1562.
Maguire, Martin (2001). 'Methods to Support Human-Centred Design'. In: *International journal of human-computer studies* 55.4, pp. 587–634.
March, Salvatore and Gerald Smith (1995). 'Design and Natural Science Research on Information Technology'. In: *Decision support systems* 15.4, pp. 251–266.
Markova, Milena, Stephanie Wilson and Simone Stumpf (2012). 'Tangible User Interfaces for Learning'. In: *International Journal of Technology Enhanced Learning* 4.3-4, pp. 139–155.
Marshall, Catherine and Gretchen Rossman (2014). *Designing Qualitative Research*. Sage publications.
Mechant, Peter et al. (2012). 'E–Deliberation 2.0 for Smart Cities: A Critical Assessment of Two "Idea Generation" Cases'. In: *International Journal of Electronic Governance* 5.1, pp. 82–98.
Mellis, David et al. (2007). 'Arduino: An Open Electronic Prototyping Platform'. In: *Proceedings of CHI Extended Abstracts*. ACM, pp. 1–11.
Miles, Matthew and Michael Huberman (1994). *Qualitative Data Analysis: An Expanded Sourcebook*. Sage.
Mingers, John (2001). 'Combining IS Research Methods: Towards a Pluralist Methodology'. In: *Information systems research* 12.3, pp. 240–259.

Mitchell, Val et al. (2016). 'Empirical Investigation of the Impact of Using Co-Design Methods When Generating Proposals for Sustainable Travel Solutions'. In: *CoDesign* 12.4, pp. 205–220. DOI: 10.1080/15710882.2015.1091894.

Mitchell, Vincent (1993). 'Industrial In-Depth Interviews: Some Considerations for First-Time Users'. In: *Marketing Intelligence & Planning* 11.4, pp. 25–29.

Murty, Rohan Narayana et al. (2008). 'Citysense: An Urban-Scale Wireless Sensor Network and Testbed'. In: *IEEE International Conference on Technologies for Homeland Security*, pp. 583–588.

Myers, Brad, Scott Hudson and Randy Pausch (2000). 'Past, Present, and Future of User Interface Software Tools'. In: *ACM Transactions on Computer-Human Interaction (TOCHI)* 7.1, pp. 3–28.

Nair, Godwin and Andreas Riege (1995). 'Using Convergent Interviewing to Develop the Research Problem of a Postgraduate Thesis'. In: *Proceedings of Marketing Education and Researchers International Conference, Gold Coast.*

Neirotti, Paolo et al. (2014). 'Current Trends in Smart City Initiatives: Some Stylised Facts'. In: *Cities* 38, pp. 25–36. ISSN: 0264-2751. DOI: 10.1016/j.cities.2013.12.010.

Nelson, Les and Elizabeth Churchill (2005). 'User Experience of Physical-Digital Object Systems: Implications for Representation and Infrastructure'. In: *Workshop on Smart Object Systems on Ubiquitous Commerce at Ubicomp.*

Olsen Jr, Dan (2007). 'Evaluating User Interface Systems Research'. In: *Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology*, pp. 251–258.

Oulasvirta, Antti and Kasper Hornbæk (2016). 'Hci Research as Problem-Solving'. In: *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pp. 4956–4967.

Parkhe, Arvind (1993). '"Messy" Research, Methodological Predispositions, and Theory Development in International Joint Ventures'. In: *Academy of Management review* 18.2, pp. 227–268.

Peräkylä, Anssi (1998). 'Reliability and Validity in Research Based on Tapes and Transcripts'. In: *Qualitative research: Theory, method and practice*, pp. 201–220.

Perlgut, Donald and Wendy Sarkissian (2005). *The Community Participation Handbook: Resources for Public Involvement in the Planning Process.* Institute for Science and Technology Policy, Murdoch University.

Pettersen, Ida Nilstad and Casper Boks (2008). 'The Ethics in Balancing Control and Freedom When Engineering Solutions for Sustainable Behaviour'. In: *International Journal of Sustainable Engineering* 1.4, pp. 287–297.

Reiersølmoen, Mikael, Francesco Gianni and Monica Divitini (2017). 'DELTA: Promoting Young People Participation in Urban Planning'. In: *Conference on Smart Learning Ecosystems and Regional Development.* Vol. 80. Springer International Publishing, pp. 77–89. ISBN: 978-3-319-61322-2. DOI: 10.1007/978-3-319-61322-2_8.

Remenyi, Dan et al. (1998). *Doing Research in Business and Management: An Introduction to Process and Method.* Sage.

Riege, Andreas (2003). 'Validity and Reliability Tests in Case Study Research: A Literature Review with "Hands-on" Applications for Each Research Phase'. In: *Qualitative market research: An international journal* 6.2, pp. 75–86.

Roberts, Nancy (2004). ‘Public Deliberation in an Age of Direct Citizen Participation’. In: *The American review of public administration* 34.4, pp. 315–353.

Robson, Colin (1993). ‘Real World Research: A Resource for Social Scientists and Practitioners-Researchers’. In: *Massachusetts: Blackwell Pushers.*

Rogers, Yvonne (2006). ‘Moving on from Weiser’s Vision of Calm Computing: Engaging Ubicomp Experiences’. In: *UbiComp 2006: Ubiquitous Computing.* Springer, pp. 404–421.

Rogers, Yvonne, Kay Connelly et al. (2007). ‘Why It’s Worth the Hassle: The Value of in-Situ Studies When Designing Ubicomp’. In: *International Conference on Ubiquitous Computing*, pp. 336–353.

Rogers, Yvonne and Henk Muller (2006). ‘A Framework for Designing Sensor-Based Interactions to Promote Exploration and Reflection in Play’. In: *International Journal of Human-Computer Studies* 64.1, pp. 1–14.

Rogers, Yvonne and Tom Rodden (2003). ‘Configuring Spaces and Surfaces to Support Collaborative Interactions’. In: *Public and Situated Displays.* Springer, pp. 45–79.

Sanders, Elizabeth and Pieter Jan Stappers (2008). ‘Co-Creation and the New Landscapes of Design’. In: *Co-design* 4.1, pp. 5–18.

Saunders, Mark, Philip Lewis and Adrian Thornhill (2009). *Research Methods for Business Students.* Pearson education.

Schuurman, Dimitri et al. (2012). ‘Smart Ideas for Smart Cities: Investigating Crowdsourcing for Generating and Selecting Ideas for ICT Innovation in a City Context’. In: *Journal of theoretical and applied electronic commerce research* 7.3, pp. 49–62.

Siegemund, Frank (2004). ‘A Context-Aware Communication Platform for Smart Objects’. In: *International Conference on Pervasive Computing*, pp. 69–86.

Sintoris, Christos et al. (2018). ‘Out of the Box: Using Gamification Cards to Teach Ideation to Engineering Students’. In: *Ambient Intelligence.* Ed. by Achilles Kameas and Kostas Stathis. Cham: Springer International Publishing, pp. 221–226. isbn: 978-3-030-03062-9.

Solanas, Agusti et al. (2014). ‘Smart Health: A Context-Aware Health Paradigm within Smart Cities’. In: *IEEE Communications Magazine* 52.8, pp. 74–81.

Streitz, Norbert et al. (2005). ‘Designing Smart Artifacts for Smart Environments’. In: *Computer* 38.3, pp. 41–49.

Tashakkori, Abbas and John Creswell (2008). ‘Mixed Methodology across Disciplines’. In: *Journal of Mixed Methods Research* 2.1, pp. 3–6. doi: 10.1177/1558689807309913.

Trigg, Randall H, Thomas P Moran and Frank G Halasz (1987). ‘Adaptability and Tailorability in NoteCards’. In: *Human–Computer Interaction–INTERACT’87.* Elsevier, pp. 723–728.

Trischler, Jakob et al. (2018). ‘The Value of Codesign: The Effect of Customer Involvement in Service Design Teams’. In: *Journal of Service Research* 21.1, pp. 75–100.

Udoh, Itorobong and Gerald Kotonya (2017). ‘Developing IoT Applications: Challenges and Frameworks’. In: *IET Cyber-Physical Systems: Theory & Applications* 3.2, pp. 65–72.

Ullmer, Brygg and Hiroshi Ishii (2000). 'Emerging Frameworks for Tangible User Interfaces'. In: *IBM systems journal* 39.3.4, pp. 915–931.

UN (2015). *Smart Cities: Regional Perspective.* URL: https://worldgovernmentsummit.org/api/publications/document?id=d1d75ec4-e97c-6578-b2f8-ff0000a7ddb6.

Vaajakallio, Kirsikka and Tuuli Mattelmäki (2014). 'Design Games in Codesign: As a Tool, a Mindset and a Structure'. In: *CoDesign* 10.1, pp. 63–77.

Van Dam, Andries (1997). 'Post-WIMP User Interfaces'. In: *Communications of the ACM* 40.2, pp. 63–67.

Vazifeh, Mohammed et al. (2018). 'Addressing the Minimum Fleet Problem in On-Demand Urban Mobility'. In: *Nature* 557.7706, p. 534.

Venkatesh, Viswanath, Susan Brown and Hillol Bala (2013). 'Bridging the Qualitative-Quantitative Divide: Guidelines for Conducting Mixed Methods Research in Information Systems.' In: *MIS quarterly* 37.1.

Von Hippel, Eric (2001). 'User Toolkits for Innovation'. In: *Journal of Product Innovation Management* 18.4, pp. 247–257.

Weiser, Mark (1991). 'The Computer for the 21st Century'. In: *Scientific american* 265.3, pp. 94–104.

Welbourne, Evan et al. (2009). 'Building the Internet of Things Using RFID: The RFID Ecosystem Experience'. In: *IEEE Internet computing* 13.3.

Wilks, Judith and Julie Rudner (2013). 'A Voice for Children and Young People in the City'. In: *Australian Journal of Environmental Education* 29.01, pp. 1–17.

Wobbrock, Jacob and Julie Kientz (2016). 'Research Contributions in Human-Computer Interaction'. In: *Interactions* 23.3, pp. 38–44.

Yin, Robert (2017). *Case Study Research and Applications: Design and Methods*. Sage publications.

Zhai, Bing et al. (2018). 'Co-Sleep: Design for Workplace Based Wellness Program to Raise Awareness of Sleep Deprivation'. In: *arXiv preprint arXiv:1809.09846*.

Part II

Research Papers

Research Papers

P1 Gianni, Francesco, and Monica Divitini (2016) **'Technology-Enhanced Smart City Learning: A Systematic Mapping of the Literature.'** *Interaction Design and Architecture(s) Journal - IxD&A* 27:28–43.

P2 Mora, Simone, Francesco Gianni, and Monica Divitini (2017) **'Tiles: A Card-Based Ideation Toolkit for the Internet of Things.'** In: Proceedings of the 2017 Conference on Designing Interactive Systems. DIS 2017. 587–598. Edinburgh, United Kingdom: ACM.

P3 Gianni, Francesco, and Monica Divitini (2018) **'Designing IoT Applications for Smart Cities: Extending the Tiles Ideation Toolkit.'** *Interaction Design and Architecture(s) Journal - IxD&A* 35:100–116.

P4 Gianni, Francesco, Lisa Klecha, and Monica Divitini (2019) **'Tiles-Reflection: Designing for Reflective Learning and Change Behaviour in the Smart City.'** In: The Interplay of Data, Technology, Place and People for Smart Learning. SLERD 2018. *Smart Innovation, Systems and Technologies* 95:70–82. Aalborg, Denmark: Springer, Cham.

P5 Mavroudi, Anna, Monica Divitini, Francesco Gianni, Simone Mora and Dag R. Kvittem (2018) **'Designing IoT Applications in Lower Secondary Schools.'** In: Proceedings of IEEE Global Engineering Education Conference. EDUCON 2018. 1120–1126. Tenerife, Spain: IEEE.

P6 Gianni, Francesco, Simone Mora, and Monica Divitini (2018) **'RapIoT Toolkit: Rapid Prototyping of Collaborative Internet of Things Applications.'** *Journal of Future Generation Computer Systems.* Elsevier.

P7 Gianni, Francesco, Simone Mora, and Monica Divitini (2018) **'Rapid Prototyping Internet of Things Applications for Augmented Objects: The Tiles Toolkit Approach.'** In: Ambient Intelligence. AmI 2018. *Lectures Notes in Computer Science.* 11249:204–220. Larnaca, Cyprus: Springer, Cham.

Paper 1

Technology-Enhanced Smart City Learning:
A Systematic Mapping of the Literature

Francesco Gianni and Monica Divitini

Interaction Design and Architecture(s) Journal – IxD&A

Interaction Design and Architecture(s) Journal - IxD&A, N.27, 2015, pp. 28-43

Technology-enhanced Smart City Learning: a Systematic Mapping of the Literature

Francesco Gianni and Monica Divitini
francesco.gianni@idi.ntnu.no, monica.divitini@idi.ntnu.no

Norwegian Univesity of Science and Technology
Department of Computer and Information Science
Trondheim, Norway

Abstract Smart cities are a popular and recognized research topic. In urban spaces, the learning factor is an important component for citizens and local communities. This paper presents a systematic mapping of the literature on smart city learning, with focus on how technology is used to enhance smart city learning. The goal is to map the state of the art and to identify gaps in current research that can prompt new research in this area.
Articles were collected from various online databases and relevant journal publications, selected according to defined inclusion/exclusion criteria. Abstracts were coded based on a number of criteria, including e.g. learning goal, used technology, and theoretical approach. Following the coding process results were analyzed to identify themes.
In the paper we shed light on the current understanding of smart city learning by (i) Identifying common scenarios and learning settings; (ii) publication patterns; (iii) technical features in the supporting technology; (iv) learning theories and approaches that are mostly used; and (v) adopted type of research and research methods.
The mapping shows that the concept of smart city learning is growing in popularity, with increasing number of publications in this area in the last years. However, the field is rather fragmented, with very different understanding of the concept. Smart city learning is also emerging as a very complex form of learning, with different stakeholders, learning activities, and technological solutions combined in rich eco-systems. The mapping also points out two largely unexplored areas of technological support, namely the Internet of Things (IoT) and the use of city-related data.

Keywords: Smart-City learning, Technology, Behavior Change

1 Introduction

The concept of smart-city has been used in many different contexts and is associated with distinctive and innovative aspects that are often quite different. Big diversities are observed on the reasons *why* different cities are defined as *smart*.

This situation is the consequence of the lack of a clear and recognized definition of smart city.

In attempting to pin down what is smart about the smart city, one finds that it involve quite a diverse range of things (information technology, business innovation, governance, communities and sustainability). It can also be suggested that the label itself often makes certain assumptions about the relationship between these things, for example regarding consensus and balance [10].

Komninos [14], in his attempt to delineate the intelligent city, (perhaps the concept most closely related to the smart city), cites four possible meanings:

1. The application of a wide range of electronic and digital applications to communities and cities.
2. The use of information technology to transform life and work.
3. The meaning of intelligent or smart as embedded information and communication technologies.
4. The spatial territories that bring ICTs and people together to enhance innovation, learning, knowledge and problem solving.

Overall then, Komninos [14] sees intelligent (smart) cities as "*territories with high capacity for learning and innovation, which is built–in the creativity of their population, their institutions of knowledge creation, and their digital infrastructure for communication and knowledge management*".

The adjective "smart" began to gain a increasingly notoriety between 2005 and 2007, when it started to be used to denote a sort of dream-city, i.e. a complex and optimized environment, or eco-system, where it could be desirable to live. It appeared immediately clear that the adjective smart was intended to go well beyond the meaning intelligent and/or to emphasize the use of ICT and digital technologies [8].

For this reason the authors chose to limit the search to articles published from 2005.

Smart cities are also a powerful and recognized ecosystem for learning. Smart city learning aim to support the improvement of all key factors contributing to the regional competitiveness: mobility, environment, people, quality of life and governance [10]. The approach is aimed at optimizing resource consumption and saving time improving flows of people, goods and data[1].

Education in this context is pursued as a bottom-up process, where person and places are central. Smartness from a learning perspective exists both in the ambient data collected and among the communities that exists within a city.

The separation between student and teacher fades out. Their role will be content or situation dependent: everybody will be a learner and the relation between persons will get a bigger role.

From the learning perspective, smart cities can be seen as an independent *learning actor* that behaves like an autonomous entity which adapt itself in an evolving environment.

[1] http://www.mifav.uniroma2.it/inevent/events/sclo/

Despite this, in this paper we focus on smart cities as a place where citizens learn smart-behaviors.

This scenario can involve traditional education which happens in facilities like schools and universities. The goal of this work is instead more oriented towards *lifelong learning*, defined as the continuous build of skills to adapt and collaborate in dynamic ecosystems like smart cities.

2 Motivation and Research Questions

Technology in smart cities is essential and considered as a supporting backbone [9]. While the role of technology in smart cities has been widely recognized and addressed, there seems to be no established field of research that connects smart cities to learning.

This work is motivated by the quest for a clear overview of existing research related to learning in smart cities.

A systematic mapping, compared to a systematic literature review, helps to spot holes in the knowledge and to get a broader overview of the topic. Since the field is rather fragmented, we decided for a mapping instead of focusing on a specific domain with a literature review.

The methods used and the process followed for the systematic mapping are described in [17].

2.1 What do we consider as "smart city learning"?

To avoid articles not relevant for the purpose of this work, the authors decided that the boundaries that define the adopted research scope on smart cities are dependent by two factors:

- The social perspective, which defines the people affected and should not be constrained by any particular bound. Every citizen can be involved.
- The urban perspective, which includes the city as an urban space and it is not confined to any particular facility or environment that can be also found outside the smart city context.

A significant scenario should include at least one of the two factors. Here are some examples of scenarios:

1. Technology supported collaboration of students with municipalities to promote participation as a form of learning: providing feedback for urban planning in public areas of the city.
2. Citizens collecting energy consumption data in their house, which is then aggregated to create a energy consumption map for the whole city. Looking at the map, citizens can discover interesting patterns and reflect on the margin of improvement for their houses.

3. Bikes used for bike sharing services can be instrumented to collect air pollution and other sensor data. Cyclists around the city can provide a detailed and constantly updated sensor-map that can stimulate citizens to adopt more sustainable and efficient mobility patterns.

All the three scenarios proposed are relevant for the smart city learning research scope defined above.

The first scenario works only within a defined community of citizens, but they are displaced in the entire urban environment of a smart city.

In the second scenario the space is confined into individual apartments and houses, but every citizen can be potentially involved. The data is also aggregated and interpreted at a city-wide level.

The third scenario combine both the social and urban perspective: there is no specific category of citizens being addressed and the relevant urban space is located in the city as a whole.

2.2 Research Questions

The research questions addressed are:

- **RQ1**: Which are the most common scenarios of application, usage settings and learning contexts within technology-enhanced smart city learning research?
- **RQ2**: Is there any characteristic publication pattern?
- **RQ3**: Which kind of features and patterns characterize the technological applications?
- **RQ4**: Which learning theories and approaches are most commonly used?
- **RQ5**: What type of research is performed and which methods are used?

3 Data Sources and Search

3.1 Data Sources

The articles were searched and collected using three different approaches:

1. keyword based search on different online databases
2. manual screening of selected conference proceedings
3. manual screening of selected journals

The following online databases were used for the keyword based search: ISI Web of Science[2], ACM digital library[3], Elsevier - ScienceDirect[4], Elsevier - Scopus[5], IEEE Xplore[6].

The following conference proceedings were searched for relevant articles:

[2] https://apps.webofknowledge.com/
[3] https://dl.acm.org/
[4] https://www.sciencedirect.com/
[5] http://www.scopus.com/
[6] http://ieeexplore.ieee.org/

- **CSCW** Computer-Supported Cooperative Work and Social Computing[7]
- **CHI** Conference on Human Factors in Computing Systems[8]
- **EC-TEL** European Conference on Technology Enhanced Learning[9]
- **AMI** International Joint Conference on Ambient Intelligence[10]
- **C&T** International Conference on Communities and Technologies[11]

The following journal issues were searched for relevant articles:

- **IJDLDC** International Journal of Digital Literacy and Digital Competence vol. 3 n. 4 - Special Issue on *"Smart City Learning, literacy and Competences"*[12]
- **IxD&A** Interaction Design and Architecture(s), vol. 16 (part I)[13], vol. 17 (part II)[14] - Special Issue on *"Smart City Learning - Visions and practical Implementations: toward Horizon 2020"*

3.2 Search and Keywords

The keywords selection process was driven by the PICO framework. PICO helps to develop a comprehensive set of search keywords for quantitative research terms according to: Population, Intervention or Exposure (PECO), Comparison, Outcomes [21].

Initially, keywords for all the sections of the framework were selected, but the authors decided later on to relax some constraints in order to avoid missing possible relevant articles. A *context* section was also added to the schema.

Table 1 shows the PICO(C) structure with associated keywords.

A pilot search was conducted on some of the online databases in order to refine the keywords and find a search query that could be adapted and used in all the different online databases.

The final search query used is reported in Table 2.

Different online databases offer different levels of search functionalities and detail when going to use a complex query that possibly involves several keywords and fields (title, abstract, etc.).

Some of the difficulties encountered were:

- limit on the number of keywords that can be used;

---

[7] http://cscw.acm.org/
[8] http://chiYYYY.acm.org/
[9] http://www.ec-tel.eu/
[10] http://www.ami-conferences.org/
[11] http://comtech.community/
[12] http://www.igi-global.com/journal/international-journal-digital-literacy-digital/1170
[13] http://www.mifav.uniroma2.it/inevent/events/idea2010/index.php?s=10&a=10&link=ToC_16_P
[14] http://www.mifav.uniroma2.it/inevent/events/idea2010/index.php?s=10&a=10&link=ToC_17_P

Table 1. PICO(C) Driven Keywords Framing

Population	-
Intervention	*learning*
Comparison	-
Outcome	*participation, collaboration, reflection, awareness*
Context	*cities, smart city, urban, connected city, intelligent city, digital city*

Table 2. Search Query

Context	(cities OR "smart city" OR urban OR "connected city" OR "intelligent city" OR "digital city")
	AND
Intervention	("learning")
	AND
Outcome	(participation OR collaboration OR reflection OR awareness)

– limit on the fields where the search can be performed, search on title AND abstract not always possible;
– no precise and direct control on the target search fields, keywords could be only searched on a preset aggregation of fields like title, abstract and article keywords;
– different ways of coding the same logic expression, the same search string couldn't be reused on different databases;
– different formats of the result set, in some cases was possible to batch-download the results, otherwise results were scraped using Zotero[15] browser integration.

The keywords were searched on title and abstract when possible, otherwise only the abstract was used. In Table 3 the size of the result sets for each online database is outlined.

The articles collected were imported in a Zotero library, and duplicates were manually removed. The complete list of selected articles is available as a public online repository[16].

---

[15] https://www.zotero.org/
[16] https://github.com/francg/IxD-A_SCL_systematic_mapping_articles
[17] Topic fields include Titles, Abstracts, Keywords and Indexing fields such as Systematics, Taxonomic Terms and Descriptors

Table 3. Result Set for online databases before duplicates removal

	ISI	**ACM**	**ScienceDirect**	**Scopus**	**IEEE**	**TOT**
n	938	35	162	1022	42	2199
field	topic[17]	abstract	abstract	abstract	abstract	

Table 4. Final Result Set without duplicates and selected articles for coding

	Online Databases	**IJDLDC**	**IxD&A**	**TOT**
n (no duplicates)	1485	5	11	1501
selected	43	2	9	54

4 Screening of Papers

After the search and collection phase, articles meta-data were exported to a spreadsheet for screening and selection of relevant topics for the study. All the titles, and if necessary the abstracts, were read to determine which articles to include in the study.

The PRISMA Statement for Reporting Systematic Reviews and Meta-Analyses [16] was used to guide and structure the criteria of inclusion/exclusion. More precisely the authors used report eligibility and study eligibility criteria.

To follow the Report and Study eligibility criteria adopted for this work.

Report Eligibility

1. Publications should be in English;
2. Articles should be published on peer-reviewed journals, international conferences or as book chapters;
3. Year of publication should be between 2005 and 2015;
4. Publications must have an abstract.

Study Eligibility

1. The city perspective must comply with the definition provided in section 2.1: the concept of city as a whole, either in the urban or citizen perspective, must be present;
2. If the object of research is a single community the study must not be limited to any urban area in the city or should be related to one of the infrastructure networks that permeates the city (streets, water and power lines, etc);
3. The learning factor should be present;
4. If the environment is limited to a specific context, there should be no constraints on categories of citizens that are involved or take advantage of the research;
5. The use of technology should be present and mentioned in the abstract.

The inclusion/exclusion screening was initially performed by both authors independently on the first 100 articles. On a total of 16 articles there was disagreement, and a specific discussion on the abstract was needed to reach a final decision of inclusion or exclusion.

This process was helpful to discuss, clarify and refine the criteria of inclusion/exclusion.

The following step consisted of another independent screening of 100 articles, this time the authors disagreed only on 4 articles. This two-step process helped to ensure that both the authors applied the inclusion/exclusion criteria in the same way, and allowed for the rest of the articles to be divided between the authors for the inclusion decision. Each author decided independently for inclusion/exclusion for the 50% of the remaining articles.

The total number of included articles is 54, while the articles excluded after title/abstract screening are 1447.

No articles were included from manual search of conference proceedings.

5 Classification and Coding

A first classification structure was drafted and used by one of the authors to code the first 20 abstracts. The coding of these abstracts and the classification were then discussed and revised by both authors.

The classification structure was created in Nvivo[18] and was organized in two nested levels.

The authors decided to use an *emerging* approach when working on the categories: new elements were dynamically added during the coding process.

The coding process itself consisted in reading the abstract and *tagging* relevant chunks of text with one or more categories.

The number of categories that could be correlated to any single publication was strictly connected with the richness and accuracy of the abstract. More information-rich and structured abstracts were tagged with more categories than shorter ones.

6 Results and Findings

RQ1: Scenarios and learning contexts

In Figure 1 the distribution of the encountered scenarios is presented. Most research is connected to schools and governance. This is confirmed by the fact that the target population of the studies and/or the community affected by the learning process are often the students.

Several publications present a scenario where students are involved in the development of an architectural or urban planning project in the city, e.g. [23],[25],[3].

This result can be correlated to the finding that more than 68% of the articles present the concept of the city as a place where the learning experience happens. This point of view is quite distinct to the more engaging concept of actively

[18] http://www.qsrinternational.com/

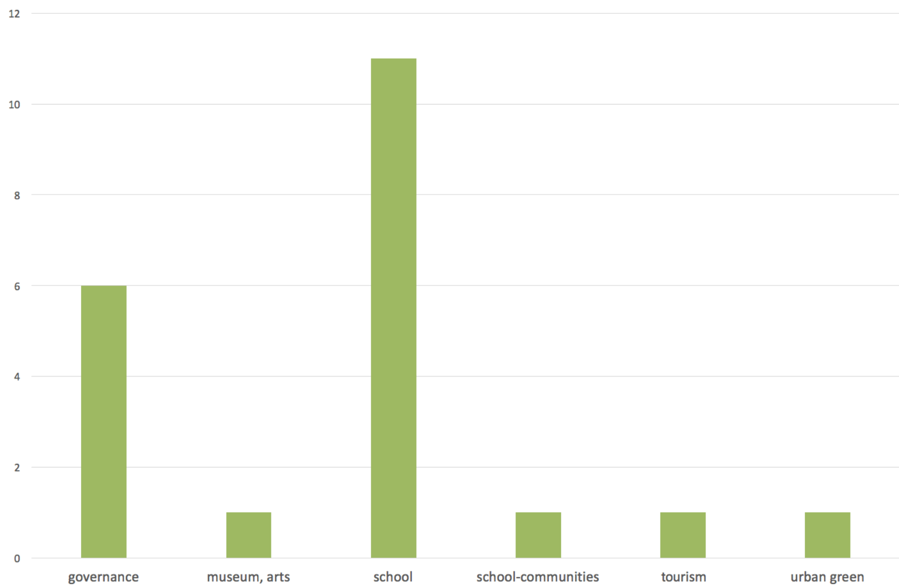


Figure 1. Identified research scenarios.

living the city, learning behaviours and generating knowledge, which can be considered a lifelong learning experience to improve the quality of urban living. As an example of this concept, some articles focus on promoting and teaching environmental-friendly practices like reducing the carbon footprint [6] or reduce dependence on owned cars to satisfy mobility needs [26].

RQ2: Publication pattern

Research on smart city learning gained approval and popularity quite constantly during the years. In Figure 2 the number of publications per year is showed. A relatively important amount of articles dates back to 2005, starting year of the chosen interval.

From 2006 a general increase in publications can be noted till 2014. The year 2015 was excluded from this statistic since not all research on the topic was yet published when articles were collected.

Selected articles are almost equally divided between international conference proceeding publications and journals or book chapters. Publications from international journals is the most numerous category.

RQ3: Application of technology

The technological pattern involved in smart city learning is, most of the times, connected to supporting the learning process, as visible in Figure 3 where the

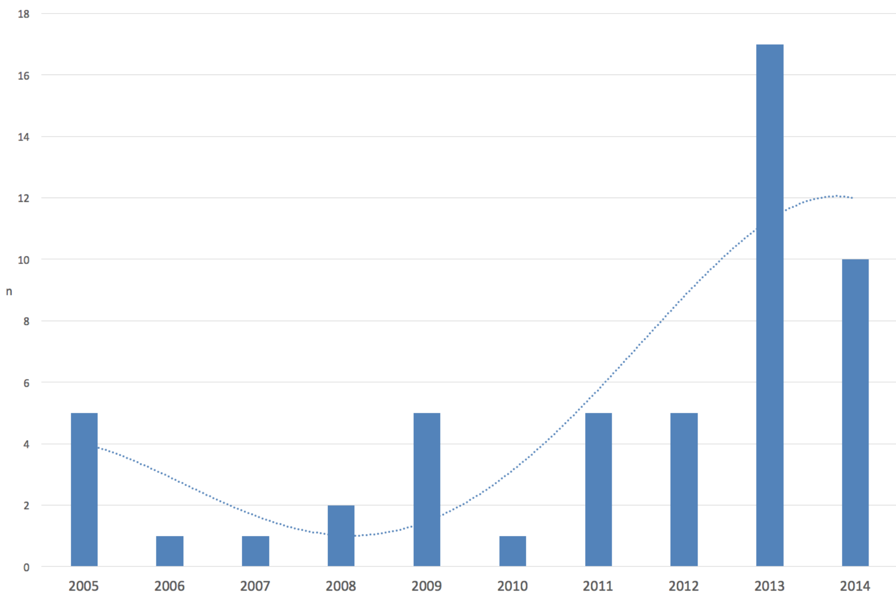


Figure 2. Research publications per year. In dots: fourth order polynomial trendline.

frequency of the identified technological patterns is presented. This means that technology is used as a tool to support a learning process that remains often structured as a traditional one. Some of the others applications use technology to acquire and collect specific information like sensor data or geographical location. From Figure 4, which gives an overview of technologies used, mobile devices emerge as the prevailing category. As an example, studies adopt mobile technologies to generate and collect data [18],[1], support language learning or others school topics and subjects [7] and as supporting technology in situated games in the city [1],[11].

Online cooperative platforms of various types are also used in many cases: more precisely e-learning [22],[12] and e-government solutions [27],[4] were mentioned in more than one article.

RQ4: Learning theories and approaches

Some articles mentioned specific learning theories applied during the study. Game-based Learning and Situated Learning [2] are the approaches that were reported more often. Figure 5 pictures which learning theories are mentioned more often in the studies. It is also worth mentioning that most of the abstracts do not mention explicitly a specific learning theory.

The articles that use Game-Based Learning methods and serious games [19], often locate the gameplay in the urban space and make use of mobile devices [11].

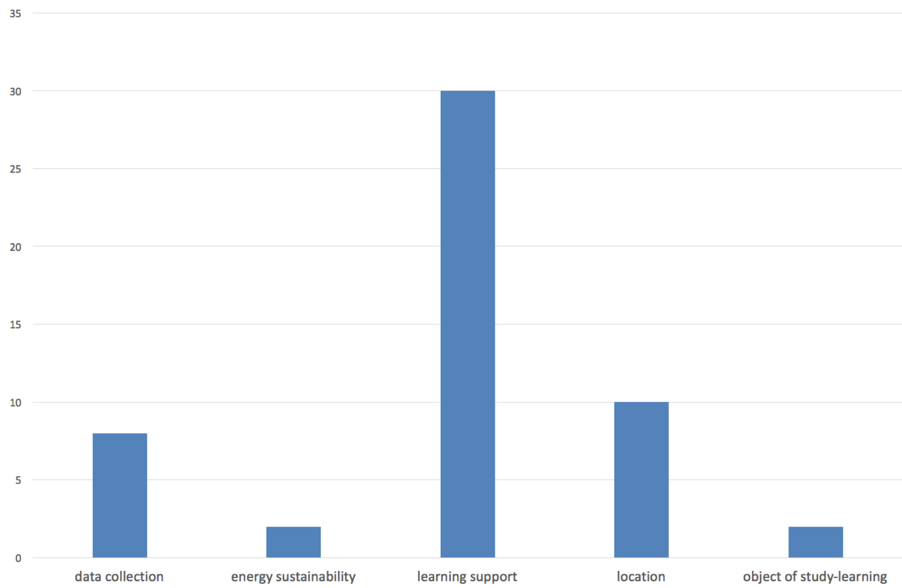


Figure 3. Identified technological patterns.

Figure 6 describes some of the pursued approaches and concepts found in the abstracts screened. The most referenced ones are connected to various levels of collaboration and cooperation between stakeholders or within the learning community (for example among the students). Context awareness and situatedness are also mentioned in a few articles.

Collaboration is crucial since many articles involve different stakeholders in the learning process, like universities and technical schools [15], decision maker, citizens and universities [6] or stakeholders located in different countries [20],[24].

RQ5: Research methods and types of research

Research on smart city learning is often limited to perform a specific problem investigation. Studies that aim to design a concrete solution or a technological implementation are less common. This means that technology, although present, is not exploited for its full potential as a facilitator instrument but rather as a marginal supporting tool.

Even more rare are studies that make use of IOT, ubiquitous technologies and custom hardware prototyping.

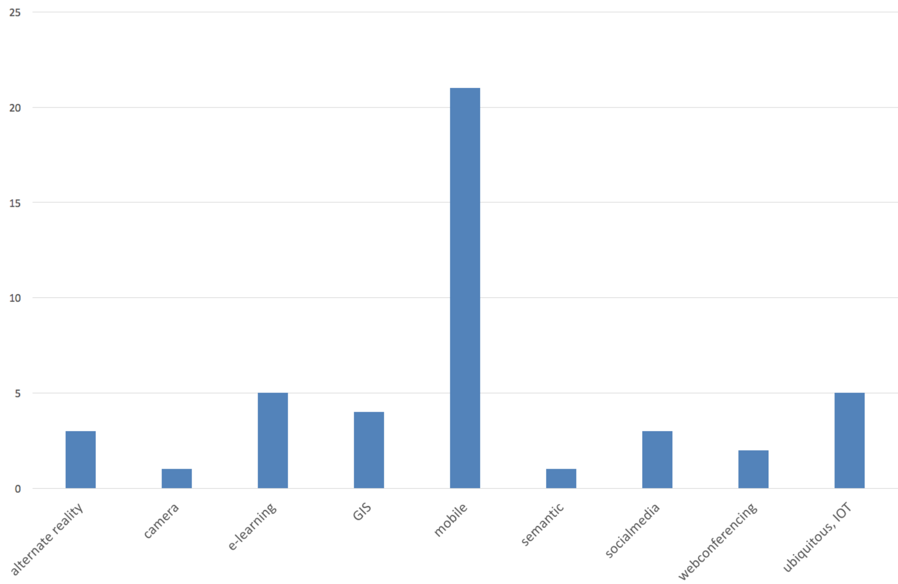


Figure 4. Technology use.

7 Discussion

Smart city learning as a multi-faceted concept

As for the concept of smart city, smart city learning is an overloaded concept. Though the term smart city learning has gained popularity, it seems there is no general understanding of the concept. As for many other terms, it does not seem to make sense to aim at a precise definition since its power is in creating an overlying umbrella.

The complexity of smart city learning

Smart city learning is emerging as a rather complex endeavor, that is challenging the way we think about technology-enhanced learning. Complexity of smart city learning is emerging along multiple dimensions.

– *Stakeholders*. Though most of the research that we have identified is actually initiated in school context, the learning process generally involves the cooperation of different stakeholders, e.g. public sectors, other citizens, domain experts. The cooperation of people with different interests and competencies creates a richer learning space, but at the same time it leads to learning processes that are more difficult to shape and to coordinate.
– *Activities*. Most of the examples of smart city learning presented in the literature include a combination of activities, often bridging formal and

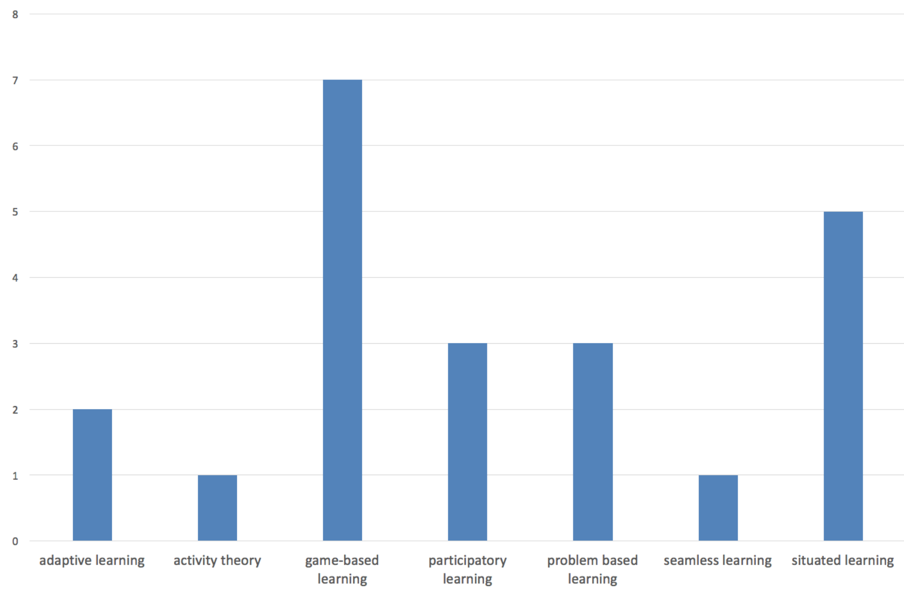


Figure 5. Learning theories applied in the studies.

informal learning. These might include, for example, data collection and generation of data in-situ, co-located and distributed processes of sense-making, participation in complex city processes, like urban planning, and sharing of knowledge within different communities. All these activities require different competencies, skills, and assessment criteria.

– *Technologies.* Smart city learning is often enhanced by different technologies, either dedicated or general purpose, like for example social media and sharing platforms. More than large institutional systems, like e.g. dedicated learning management systems, the field seems to be characterized by a tailored adoption of multiple lightweight systems. In addition, the success of the learning experience is relying on the availability of a technical infrastructure to promote communication.

As a consequence of this complexity we need to re-think our research methods and design processes to meet the specific challenges of this new domain. Smart city learning is happening in complex eco-systems that require new theoretical approaches, multidisciplinary approaches, and new pedagogics. A literacy of participation needs to be developed.

Unexplored technical opportunities

In the mapping we have identified a number of interesting concepts and technological solutions. However, we have also identified two technical opportunities that are, somehow unexpectedly, not yet used.

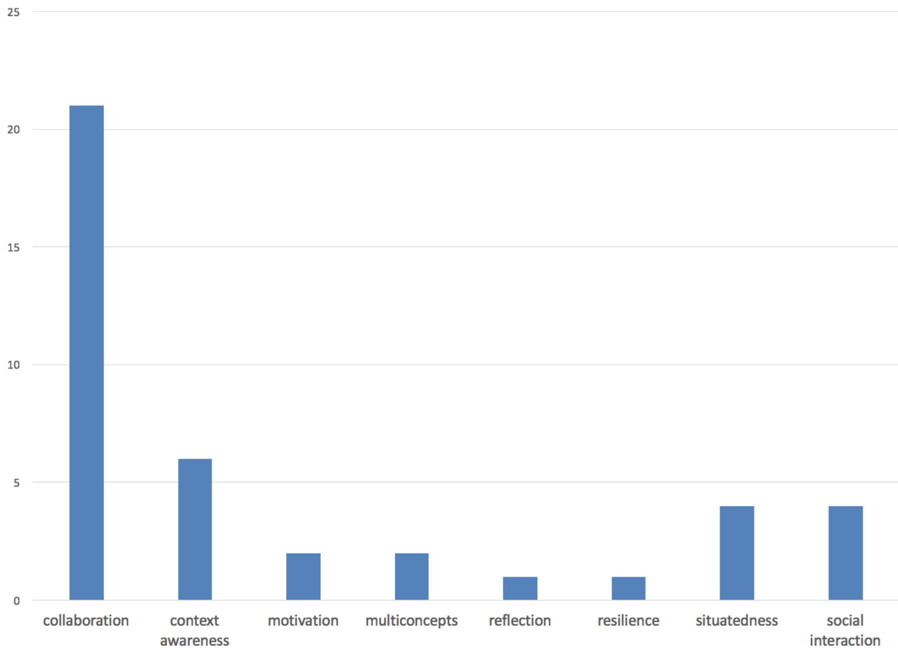


Figure 6. Pursued approaches and concepts.

– The proposed solutions are mostly relying on mobile technologies, e.g. phones. Novel interaction modalities, e.g. interactive objects and the Internet of Things (IoT), are not fully exploited. Some works exploit the possibility to tag objects with RFID for situated access to information [13]. These works show how novel technological solutions could be exploited to situate more learning activities. Novel interaction modalities might also support interaction with new categories of users, e.g. the elderly.
– *Big data, real-time data, small-data.* Smart cities are characterized by technical infrastructures that produce rich datasets, e.g. about mobility, energy consumption, environmental data. The solutions that are currently proposed in the literature are not fully exploiting the possibilities offered by city-related data. In the wider research area of Human Computer Interaction, the use of city data is for example used to increase awareness of environmental issues, see e.g. [5]. At the same time, research in the area of the quantified-self is taking advantage of data generated by each individuals to promote e.g. sustainable behavior. Bringing together the quantified city with the quantified self might lead to new interesting opportunities for learning, especially in the area of sustainable behavior. Of course, the use of data, especially big data, comes with a number of risks and ethical concerns, but investigating its use for social innovation and learning seems to be a promising area of future research.

8 Conclusions

The mapping of the literature was challenging because this is an emerging field, rather fragmented and characterized by a terminology that has not yet stabilized. Also, compared with other research fields that have developed more standardized ways to formulate abstracts, in this field the quality of the abstracts varies. Consequently, it varies the quality and quantity of information that can be extracted. Being aware of these challenges, we have combined manual and automatic searches of the literature to assure the coverage of a significant portion of relevant literature. We have also carefully analyzed abstracts to extract as much information as possible.

As part of our future research we aim at performing systematic literature reviews to shed light on the identified gaps. At the same time, we are also starting our design activities towards the development of a framework for supporting design and development of applications for smart city learning.

References

1. Akkerman, S., Admiraal, W., Huizenga, J.: Storification in History education: A mobile game in and about medieval Amsterdam. Computers and Education 52(2), 449–459 (2009)
2. Anderson, J.R., Reder, L.M., Simon, H.A.: Situated Learning and Education. Educational Researcher 25(4), 5–11 (Jan 1996)
3. Beckett, K., Shaffer, D.: Augmented by reality: The pedagogical praxis of urban planning as a pathway to ecological thinking. Journal of Educational Computing Research 33(1), 31–52 (2005)
4. Deakin, M.: Intelligent cities as smart providers: CoPs as organizations for developing integrated models of eGovernment Services. Innovation-the European Journal of Social Science Research 25(2), 115–135 (2012)
5. Dourish, P.: HCI and Environmental Sustainability: The Politics of Design and the Design of Politics. In: Proceedings of the 8th ACM Conference on Designing Interactive Systems. pp. 1–10. DIS '10, ACM, New York, NY, USA (2010)
6. Evans, J., Karvonen, A.: 'Give Me a Laboratory and I Will Lower Your Carbon Footprint!' - Urban Laboratories and the Governance of Low-Carbon Futures. International Journal of Urban and Regional Research 38(2), 413–430 (Mar 2014)
7. Gaved, M., Luley, P., Efremidis, S., Georgiou, I., Kukulska-Hulme, A., Jones, A., Scanlon, E.: Challenges in Context-Aware Mobile Language Learning: The MASELTOV Approach. Mobile as Mainstream-Towards Future Challenges in Mobile Learning, Mlearn 2014 479, 351–364 (2014)
8. Giovannella, C.: Smart Learning Eco-Systems:"fashion" or "beef"? Journal of e-Learning and Knowledge Society 10(3) (2014)
9. Giovannella, C.: Smart Territory Analytics: toward a shared vision. In: 47th SIS Scientific Meeting of the Italian Statistical Society. pp. 10–14 (2014)
10. Hollands, R.G.: Will the real smart city please stand up? City 12(3), 303–320 (Dec 2008)
11. Huizenga, J., Admiraal, W., Akkerman, S., ten Dam, G.: Cognitive and Affective Effects of Learning History by Playing a Mobile Game (2008)

12. Kabaka, M., Stoltenkamp, J.: E-Learning Tools for Public Awareness Programme Education in Disaster Risk Management: Case Study of the City of Cape Town Disaster Risk Management Centre. Proceedings of the 8th International Conference on E-Learning pp. 488–495 (2013)
13. Kashtan, Y., Forkosh-Baruch, A.: Outdoors Virtual Museum. Edulearn13: 5th International Conference on Education and New Learning Technologies pp. 275–281 (2013)
14. Komninos, N.: Intelligent Cities: Innovation, Knowledge Systems, and Digital Spaces. Taylor & Francis (2002)
15. Lee, W.J., Blyden, B.: A platform for information dissemination of MicroGrid technologies. In: IEEE Power and Energy Society General Meeting (2011)
16. Liberati, A., Altman, D.G., Tetzlaff, J., Mulrow, C., Gøtzsche, P.C., Ioannidis, J.P., Clarke, M., Devereaux, P.J., Kleijnen, J., Moher, D.: The PRISMA Statement for Reporting Systematic Reviews and Meta-Analyses of Studies That Evaluate Health Care Interventions: Explanation and ElaborationPRISMA: Explanation and Elaboration. Annals of Internal Medicine 151(4), W–65 (Aug 2009)
17. Petersen, K., Feldt, R., Mujtaba, S., Mattsson, M.: Systematic mapping studies in software engineering. In: 12th International Conference on Evaluation and Assessment in Software Engineering. vol. 17. sn (2008)
18. Philip, T., Schuler-Brown, S., Way, W.: A framework for learning about big data with mobile technologies for democratic participation: Possibilities, limitations, and unanticipated obstacles. Technology, Knowledge and Learning 18(3), 103–120 (2013)
19. Poplin, A.: Digital serious game for urban planning: "B3-Design your Marketplace!". Environment and Planning B-Planning & Design 41(3), 493–511 (2014)
20. Ross, E.: Facilitating Multi-Disciplinarity, Cross cultural Collaboration in Architectural and Urban Design (2009)
21. Schardt, C., Adams, M.B., Owens, T., Keitz, S., Fontelo, P.: Utilization of the PICO framework to improve searching PubMed for clinical questions. BMC Medical Informatics and Decision Making 7(1), 16 (2007)
22. Schneider, G., Arnoldy, P., Mangerich, T.: A location based e-learning system. In: Webist 2007 - 3rd International Conference on Web Information Systems and Technologies, Proceedings. vol. SEBEG, pp. 527–531 (2007)
23. Seitamaa-Hakkarainen, P., Kangas, K., Raunio, A.M., Viilo, M.: Architecture Project: City Plan, Home and Users - Children as Architects. In: Ruismaki, H., Ruokonen, I. (eds.) 5th International Conference of Intercultural Arts Education 2012: Design Learning, vol. 45, pp. 21–31 (2012)
24. Severengiz, M., Seidel, J., Steingrímsson, J.G., Seliger, G.: Enhancing Technological Innovation with the Implementation of a Sustainable Manufacturing Community. Procedia CIRP 26, 52–57 (2015)
25. Ulrich, C., Nedelcu, A.: Let's Play as Architects in the City! Use of Mobile Technologies During the Pilot Phase. In: Roceanu, I., Stanescu, I., Barbieru, D. (eds.) Quality and Efficiency in E-Learning, Vol 2, pp. 167–172 (2013)
26. Valle, G., Apolloni, B., Epifania, F.: Cloud design for learning and mobility services. In: Proceedings: DMS 2011 - 17th International Conference on Distributed Multimedia Systems. pp. 181–184 (2011)
27. Wong, C., Deakin, M., Isacker, K., Allwinkle, S.: A prototype km system for urban regeneration: The case of intelcities IP. In: Queensland University of Technology Research Week International Conference, QUT Research Week 2005 - Conference Proceedings (2005)

Paper 2

Tiles: A Card-Based Ideation Toolkit for the Internet of Things

Simone Mora, Francesco Gianni and Monica Divitini

In: Proceedings of the 2017 Conference on Designing Interactive Systems (DIS)

DOI: https://doi.org/10.1145/3064663.3064699

Paper 3

Designing IoT Applications for Smart Cities: Extending the Tiles Ideation Toolkit

Francesco Gianni and Monica Divitini

Interaction Design and Architecture(s) Journal – IxD&A

Interaction Design and Architecture(s) Journal - IxD&A, N.35, 2017, pp. 100-116

Designing IoT Applications for Smart Cities: extending the Tiles Ideation Toolkit

Francesco Gianni and Monica Divitini
francesco.gianni@ntnu.no, monica.divitini@ntnu.no

Norwegian University of Science and Technology
Department of Computer and Information Science
Trondheim, Norway

Abstract The internet of things (IoT) is gaining momentum as a technical tool and solution for a diverse range of societal challenges. These challenges include smart cities sustainability issues which are widely recognized by decision makers and societies. Despite this, few works try to tackle these challenges empowering citizens through IoT technologies. In this paper we describe how the Tiles toolkit, a card based idea generation toolkit for IoT, has been extended to support non experts in creating ideas addressing societal challenges that affec modern smart cities. We briefl introduce the Tiles generic toolkit, then we describe in detail the extensions proposed on the cards, cardboard and how the new components are employed in a refine workshop protocol. We report the results obtained during a fiel study of the extended toolkit, where several groups of students collaborated to generate ideas involving IoT in the smart city. We discuss success and failures, drawing our conclusions after analyzing quantitative and qualitative data collected during the workshop. We conclude the article reporting the lessons learned, critical considerations about our experience evaluating the extended toolkit and reflection on possible improvements for future works.

Keywords: Smart-City learning, IoT, Design, Behavior Change

1 Introduction

The concept of smart-city has been used in many different contexts and is associated with distinctive and innovative aspects that are often quite different. Big diversities are observed on the reasons *why* different cities are defined as *smart* [6].

With his work, Komninos [10], tries to delineate the intelligent city, perhaps the concept most closely related to the smart city. Four are the possible meanings: (i) the introduction of a wide range of electronic and digital applications to communities and cities, (ii) the use of information technology to transform life and work, (iii) the meaning of intelligent or smart as embedded information and communication technologies, (iv) the spatial territories that bring ICTs and people together to enhance innovation, learning, knowledge and problem solving.

Despite this, in this paper we focus on smart cities as a place where citizens learn smart-behaviors supported by an ideation and rapid prototyping toolkit.

This scenario can involve traditional education which happens in facilities like schools and universities. The goal of our work is instead more oriented towards *lifelong learning*, defined as the continuous build of skills to adapt and collaborate in dynamic ecosystems like smart cities.

To achieve that, we envision the involvement of non-experts in design and prototyping of pervasive Internet of Things (IoT) applications for smart cities. Citizens are an example of *non-experts in IoT*, which we define as users that differ from professionals by the fact that they do not have any skill in electronic, networking protocols or assembly and configuration of IoT devices. They don't need either to be familiar with IoT and its definition.

The idea at the heart of IoT is that all the things and all the environments can be improved from a functional point of view via the embedding of technology that remains invisible to the eye of the users, which enables both products and environments to become smart [19]: meaning that they collect data from their surrounding producing high-resolution data [3], as well as communicating among themselves and with the humans, building ecologies of smart devices intertwined with urban communities and citizens.

Since the term Internet of Things was coined in 1999 [1], research has mainly focused on Wireless Sensor Networks (WSN) and Machine-to-Machine (M2M) systems. Few works have taken into consideration HCI theories and user involvement in the design of IoT applications [11]. WSN and M2M applications do not allow end users to directly interact with the technology, which is empowered mainly for data collection and remote sensing purposes. Yet, we foresee IoT as the enabling technology for ecologies of interconnected smart objects. These connected objects retain their original appearance but are augmented with technology to gain sensing capabilities and interaction properties.

In this paper we present an extension to the Tiles Ideation Toolkit (hereafter abbreviated as Tiles) [13], a card-based design toolkit for IoT user experiences. Design is seen as a matter of generating ideas then testing them, modifying and improving where necessary. Thus ideation – the formulation of initial ideas and thoughts as both personal and collaborative processes – is embraced as an enabling factor in design practices [4]. With Tiles, we only focus on applications where the user is kept *in the loop*, through tangible interaction and physical manipulation of augmented objects.

Tiles comprises a set of 110 cards and a workshop protocol to engage non-experts in idea generation. Providing non-experts only with a set of cards can be overwhelming and confusing. Browsing the cards without any guidance or constraint might not be a sufficient stimulus for creative and collaborative thinking [8]. For these reasons, Tiles provides an ideation technique and a set of workshop-related tools: (i) a cardboard, that scaffolds the use and placement of the cards, facilitates group collaboration, and contains a storyboarding and reflection phase, (ii) a playbook to guide the users step-by-step in the ideation process, (iii) user-centered design artifacts such as *personas* and *scenarios*, to address specific problem domains.

In this paper, we further specialize the Tiles approach towards applications for smart cities. We aim at raising citizen awareness, facilitating lifelong learning and providing a tool to scaffold ideas to tackle societal challenges affecting modern cities.

Current research on smart cities presents a technological gap: few studies make use of IoT and smart objects [6]. Extending the Tiles toolkit we aim at establishing a first point of contact, exploring the solution space offered by IoT and smart objects in the context of smart cities. A recent review of the literature demonstrated that in the context of smart cities, citizens are not sufficiently involved in designing and implementing technological applications [9]. Their involvement in the studies, if present, is often relegated to a tokenistic role.

Participatory design is defined as a set of theories, practices, and studies related to end-users as full participants in activities leading to software and hardware computer products and computer-based activities [7]. Active user involvement in the ideation, design and development process through participatory design or others co-design techniques, is still scarcely adopted in the smart city domain.

2 Related Work

In this section we report related works on design, brainstorming and ideation tools for smart cities. An overview of card-based ideation toolkits for the IoT is provided on [13], where the generic Tiles Ideation Toolkit is also presented and described. Design thinking workshops are often used to scaffold and support brainstorming and cooperative practices. In [14], various ideation tools are used during a smart city workshop involving different stakeholders. Ideas are categorized using post-its and personas, with the ultimate goal of enabling energy integration into urban design and promoting energy consumption awareness in the citizens.

Wagner et al. [20] employ user-centered design methods for urban planning in their 'MR tent' study. Among the other tools, in their workshops they use scenarios and 'content cards', arranged on a whiteboard and used as placeholders for different urban objects and infrastructures. Schuurman et al. [16] adopt crowdsourcing as a method to brainstorm ideas for various scenarios related to smart cities. They used an online platform to collect and vote the ideas generated by the citizens. The ideas were also briefly evaluated using a combination of feasibility and originality criteria. The ideas were divided into the predefined categories of e-government, housing, mobility, security, sport & recreation and other. In a similar way, Mechant et al. [12] compared two crowdsourcing platforms to collect ideas for smart cities. The contributions of the users addressed several facets of the city, while the most popular idea categories were 'smart mobility', 'guiding applications' and 'social bonding applications'.

In their attempt to create a framework for co-design in smart cities, Fu and Lin [5] employed brainstorming workshops for the ideation phase, and rapid prototyping for the implementation phase. Their ultimate goal was to facilitate the research, design and prototyping of a range of specialized products for new urban lifestyles. They also adopted personas and scenarios, but their use was limited to demonstrate possible solutions and to illustrate the findings.

All the smart city works presented employ some kind of ideation or brainstorming instrument to support the creative process, or at least mention that a part of the study was dedicated to generate and collect ideas. The main differences between the extended Tiles toolkit and these works can be summarized in:

1. **Timeframe** – the Tiles workshop is structured to generate and refine an idea in less then two hours of group work, regardless of the level of expertise in IoT or smart cities the users may have;
2. **Focus on smart cities** – the smart city extension of Tiles targets specific societal challenges of the city;
3. **Defined structure** – the ideation process is precisely structured, a step-by-step playbook and a cardboard allow the users to work in autonomy without direct supervision by researchers;
4. **Guidance** – the workshop is structured to encourage the users to design for a pre-determined persona and scenario, providing some guidance without limiting creativity, users are in any case free to provide their own scenario and persona;
5. **IoT oriented** – although having an idea as direct outcome, the Tiles workshop is oriented towards rapid prototyping of IoT applications, the idea generation process is tightly connected to the technical implementation.

As a comparison, the 'MR tent' study is a complex brainstorming workshop that took several days of work and multiple creative instruments to produce an idea. In their study, Schuurman et al. focused on collecting the ideas, without providing guidance in the actual process of idea generation. None of the studies presented is clearly structured towards prototyping with IoT technology or through other technologies.

3 Extending the Tiles Ideation Toolkit

3.1 The Tiles Toolkit

The original Tiles toolkit is intended to be generic, no particular focus on any application domain is provided, although the IoT role remains central. Tiles is composed of three essential elements:

1. **Cards** – a total of 110 cards organized in 7 decks, as shown in Fig. 1 each deck has a specific name and color;
2. **Cardboard** – shown in Fig. 2, is printed on a big paper roll sized 0.6 x 1.2 meters, it holds the cards during the idea generation process and includes a storyboard depicting a use case of the idea;
3. **Workshop protocol** – consisting in a step-by-step playbook which guides the users in the idea generation process, explaining how and when each deck of cards is intended to be employed.

Figure 1. The Tiles card decks composing the original toolkit, with a sample card for each deck on the bottom.

The workshop starts with the selection of an arbitrary number of everyday objects, represented in the *things* cards. The objects are chosen by the users based on the perceived usefulness, in relation to the idea and the problem addressed. These objects are then *augmented* through the addition of sensing and actuation capabilities: *services* and *human actions* cards allow to trigger a specific reaction when data coming from online services is received or when the object is physically manipulated by a human being. *Feedback* cards are used to specify how the object reacts when triggered. In addition, *connectors* cards can be used to indicate a condition that joins the behaviour of two or more smart objects. Finally, *missions* and *criteria* decks are used to stimulate divergent-convergent thinking and promote reflective learning. *Missions* provide creativity triggers through a set of provocative design goals, while *criteria* are composed by reflective statements that push the user to evaluate the idea from a different perspective.

3.2 Smart City Extension

To extend Tiles and specialize it for smart cities, we created several new cards, a new cardboard and a refined workshop protocol. The improved cardboard and the new cards are shown in Fig 3 and Fig. 4. The changes were also addressing minor usability issues that emerged during the first evaluations of the generic toolkit.

The new cards are domain specific for smart cities. Several *things* cards were added, representing urban furniture, public transport and objects usually found in a city. These new cards help the users to get familiar with the environment of the city, they are the first contact point and the initial trigger for the ideation process, together with *personas* and *scenarios*. The content of these initial decks of cards do not require any technical or scientific knowledge to be grasped, avoiding to overload the users with new concepts or abstraction from the very beginning. Letting the users free to browse through a set of plain objects and urban furniture is a strategy to smoothly introduce them into the augmentation process of the

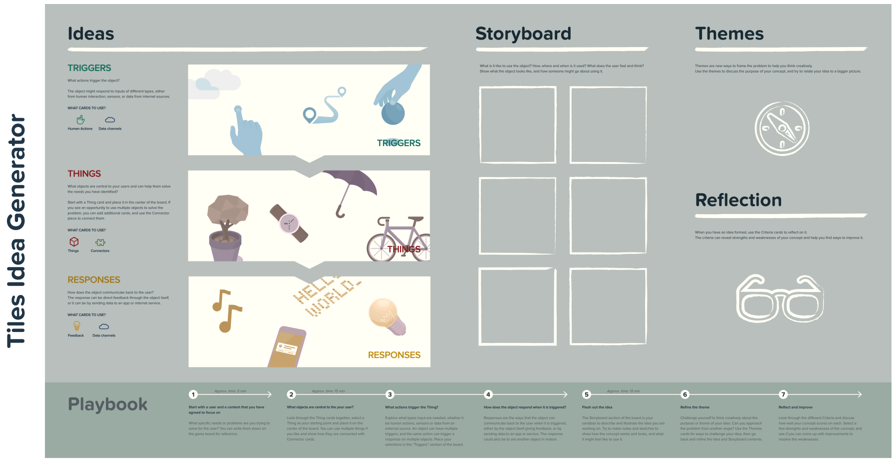


Figure 2. The Tiles original cardboard.

object(s). This allow for a progressive knowledge building process, where the more complex, augmented capabilities like ambient sensing and HCI interactions build on the top of an already established first idea seed.

A few new *feedbacks* and *human actions* were also added. Three new decks composed by 9, 10 and 5 cards were created. The first deck, *sensors*, represents sensor data from the ambient surrounding the object, like temperature, air pollution and relative humidity. These cards were meant to build awareness enabling 'augmented personal sensing', intended as the ability to provide personalized data sensing capabilities through technology. Ambient data sensing is nothing new, but it's important to take into consideration at which scale it happens. For example air pollution is known to vary considerably from street to street, while sensing stations usually operate at fixed locations, and are sparsely and strategically distributed for regulatory enforcement rather than situated to provide overall neighborhood coverage for human health [18]. Air pollution data provided by a wide area sensor network can then significantly differ from what experienced by singular citizens or urban communities. The same applies to other sources of urban data, the risk is to waste the effort of citizens trying to improve a near optimal, but unknown or misinformed, scenario while not taking action to correct a critical local situation because the city-level data are not alarming. Having an augmented object able to provide user-tailored sensor data can be an enabler and a trigger for reflective learning, motivating the users to take action effectively, accurately improving their behaviors in the city toward more sustainable ones.

The second deck is composed by 10 *personas* cards. A persona is an archetype of a user that is given a name and a face, and it is carefully described in terms of needs, goals and tasks. During the design process the design team tries to satisfy the persona's needs and goals [2]. The *personas* cards do not address only single individuals but also small groups of people like elderly people or construction

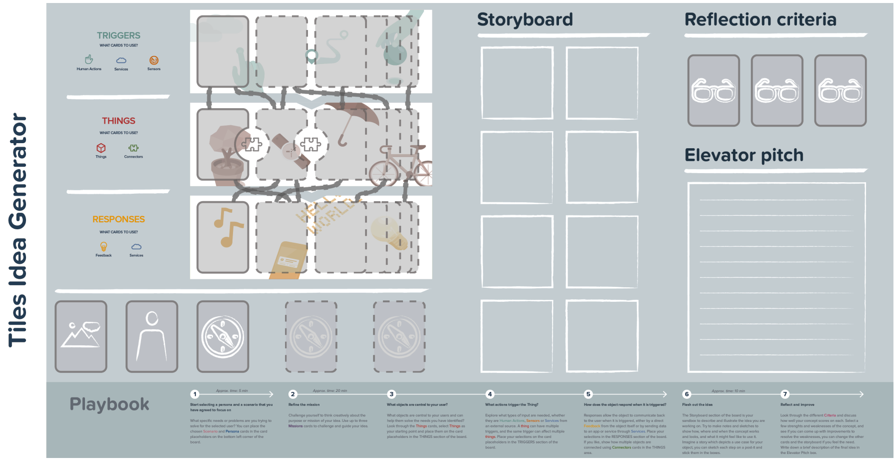


Figure 3. The new Tiles cardboard used during the workshops.

workers in the city. Personas are pushing the users to portray themselves as a particular group of citizens or individual, promoting out-of-the-box design thinking and ideation for minorities and less represented communities. The users are always free to define their own persona, but they are not allowed to start the design process without having one.

The third deck is composed by 5 *scenarios* or societal challenges affecting modern cities. Inspiration for the scenarios included came from the regional perspective report on smart cities, published by the United Nations [17] and from the sustainable development goals[1] adopted by the United Nations in 2015. More precisely, the *scenario* cards address themes included in sustainable development goals number 11, ‘sustainable cities and communities’ and 12 ‘responsible consumption and production’. *Scenarios* provide design space constraints, meant to help the users to focus the ideation process. They provide at the same time an opportunity to effectively contribute solving real world urban challenges through creative thinking and innovative technologies.

Each of the new decks adopt the same graphic style as the original cards, is color coded to be easily recognizable and contains a ‘custom card’: a blank card that can be used to add additional *sensors*, *personas* or *scenarios* directly by the users during the workshop.

The instructions on the playbook printed on the bottom of the cardboard have been updated to include the new decks. The first step became the choice of exactly one *scenario*. The users were free to pick any of the ones available in the deck. Once they decided about the problem to tackle, they were required to focus on solving it for a particular end user, chosen freely among the *persona* cards. The last new deck is the one containing the *sensors* cards. They were introduced

[1] http://www.un.org/sustainabledevelopment/sustainable-development-goals/

Figure 4. The new cards and decks composing the smart city extension, with a sample card for each deck on the bottom.

as an additional trigger element on top of *human actions* and *services*, they are all employed at the same time in the 'triggers' section of the board.

The workshop session usually started with a brief presentation introducing the users to the toolkit and the concepts of IoT and smart cities. The groups were then let free to follow the steps reported on the playbook in autonomy. Researchers and mentors supported or guided the participants only if explicitly requested by them. After the groups completed all the steps in the playbook they were encouraged to present their idea to the other participants and to the researchers, or alternatively write down the elevator pitch transcription in the apposite space on the new cardboard (Fig. 3).

We now report an example of idea generated with the extended toolkit, to better illustrate the use of the cards and the toolkit. We imagine to generate an idea for a *scenario* addressing waste management, using municipality employees as *persona*. A smart bin (*things* card) can be equipped with a tilt sensor (*human action*) and a sound alarm (*feedback*). The intended behavior might consist of the alarm sound to be triggered when the bin is detected to be falling into an horizontal position due to wind or other events. Another silent alarm is triggered when the air quality (*sensor*) surrounding the bin is compromised due to toxic emissions also causing bad smell, for example when the day is very hot. Using a *connector* card, it is possible to specify a cotemporality condition that involve a second smart object. As an example, when one of the smart bin alarms is triggered, the connector card instructs the same trigger to be used to make a smart ring vibrate, to alert a responsible person of the event. Smart objects can be triggered by multiple events and provide multiple feedbacks, the smart bin can also be connected to twitter (*service*) to keep a trace of the events occurred, providing a secondary notification channel.

4 User Study

We evaluated the extended Tiles toolkit during several workshops with university students, high school students, decision makers, researchers and professionals in urban planning.

In this section we will present the evaluation methodologies adopted, the tools employed and the design of one of the workshops. The evaluation aspects considered include the new cards, the workshop experience and the ideas generated.

4.1 Material and Tools

The participants attending the workshop were organized in groups. Each group had at disposal a cardboard, a deck of the Tiles cards including the smart city extension, post-its of different colors and markers or pens. At the end of the workshop a questionnaire was distributed to each participant. A digital camera and a camcorder were used by the authors to document the process and the produced artifacts.

4.2 Design

The workshop participants were first year university students in computer science. The workshop took place as part of a university course in IoT, replacing one of the first lessons. The category of participants matched the definition of non-experts in IoT and at the same time provided a group of users motivated to learn more about IoT. 60 students participated to the workshop, divided into 16 groups. Every user participated to only one workshop. The average number or users per group was 4. A detailed overview of users and groups is provided in Table 1.

Table 1. Details of the workshop analyzed.

workshop	N	groups	age	occupation	date
W10	60	16	19-27	university students	15/02/2017

To start, the participants were briefly introduced to the concepts of IoT as object augmentation, smart cities and to the cards and cardboard. They were then let free to browse the cards and start following the ideation process, as indicated in the playbook printed on the cardboard. After 40 minutes the ideation process was concluded and each group presented the idea during a 60 seconds elevator pitch.

During the workshop, the authors were available to support the participants if they were in need or asked for help, notes on the observed user behaviours were collected at the same time. Pictures of the cardboard, storyboard and the cards were collected at the end of the workshop, as well as video recordings of the idea pitch. Finally, every attendant compiled a questionnaire using a five steps Likert scale.

4.3 Methodology

Based on the amount and nature of collected information, for the purpose of this paper we used both qualitative and quantitative data analysis.

A sample of the ideas generated was examined to understand and evaluate how the users employed the cards and how coherent the ideas were with the cards used. We chose the ideas based on the originality, quality and clarity of the pitch, variety of the cards used and role of IoT in the concept. More precisely, we took into account the following: (i) the presence of some type of augmented object, possibly excluding common technologies like smartphones and smart watches, (ii) the involvement of more connected objects to form an ecology of smart devices, (iii) the actual use of the objects as tangible interfaces, and not only as sensor probes, (iv) the novelty of the concept, which ought to be someway different to existing applications or solutions. The themes of object augmentation and IoT are fundamental and it was important to confirm that they did not get lost during the ideation process.

To understand if the newly introduced cards were perceived as useful and contributed to support the user in the ideation process, we analyzed the pictures of the final cardboard, paying attention to how and where the new cards were used. *Scenario* and *persona* cards were excluded from the analysis since they were mandatory to use in quantity of one each, and addressed in the very first steps of the playbook. We focused the analysis on the smart city related *things* cards and the new deck of *sensors* cards.

The statistics considered in the evaluation are:

1. How many users utilized the new smart city cards from the *things* deck;
2. How many users utilized cards from the new smart city deck of *sensors*.

Data from the questionnaires was used to understand if the users enjoyed the ideation process and if they believed that the smart city version of the Tiles toolkit helped them in creating new ideas for IoT applications. We will present the results of five of the questionnaire statements, covering perceived enjoyment, ease of use, guidance and usefulness.

5 Results and Findings

We present in this section the results of the user study, combining both quantitative data and qualitative observations.

5.1 Observed Attitude

During the workshop, some of the users were stressed about the short amount of time at disposal for the ideation session, limited by the rigid schedule of the university lectures. This sentiment of stress and frustration is clearly detectable in the questionnaires. However, even if they were forced into a fast pace of work, all the groups were able to successfully follow the complete set of steps in the playbook. All the groups succeed generating an idea in 40 minutes, a very limited amount of time, and finally pitched it to the other participants. It is clear that some users were more susceptible and sensitive to the stressful condition than others, which better tolerated working under pressure.

A limited number of participants expressed concerns about the level of constraints imposed by the workshop protocol. The comments were pointing in two opposite directions, but were equally balanced in number: half of the users were complaining about having too many constraints imposed, while the other half was lamenting having too few.

5.2 Cards

To evaluate if the newly introduced cards were perceived as useful, we analyzed how often they were used. Since the users were not obliged to use any of them, our assumption is that when they did, they considered the new cards as more helpful or fit for the idea being developed.

The new smart city cards in the *things* deck were used by 50% of the users, while cards from the *sensors* deck were used by 60% of the users.

5.3 Idea Generated

We now report two ideas that have been generated during the workshop.

The first idea targets disabled people in wheelchair which make use of public transportation. The concept empower an augmented bus stop and a smart key ring. Thanks to gps sensors and visual recognition, the bus driver is automatically notified when he is approaching a bus stop where a wheelchair user is waiting. Ramps and other accessibility measures can then be deployed without delays. Once on the bus, disabled persons can tap their smart key ring near the door to communicate that they will need to hop off at the next stop. The bus will automatically deploy the ramps since the stop was reserved using the key ring, and not the usual button. The cards employed, the storyboard and cardboard are reported in Fig. 5.

The *scenario* card used was addressing the need to increase participation in public life, improving awareness of urban activities among the citizens. The *persona* chosen by the group was Tom, a 43 years old disabled man living on a wheelchair.

The storyboard is composed of 8 illustrations and contains many of the elements represented on the cards employed, like the target user on wheelchair,

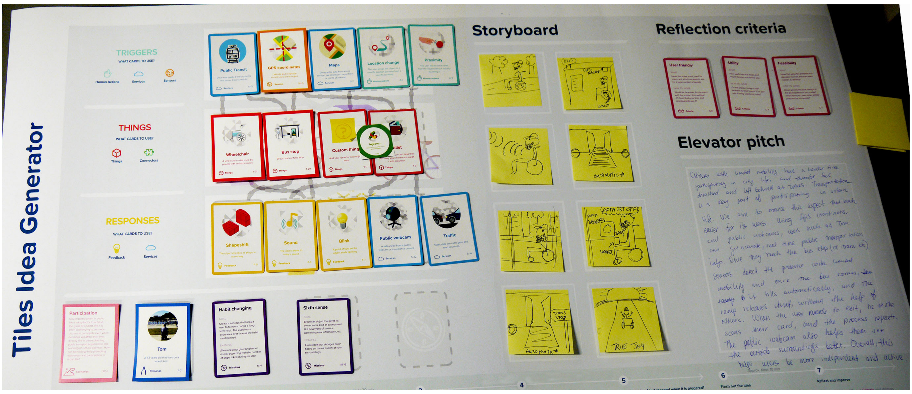


Figure 5. Cardboard and cards of the firs idea.

the bus stop, the webcam, the online data service on public transport and the triggers of location change and proximity.

The second idea redefines the concept of 'dead man switch' in the context of construction sites. Electric hand tools employed by the workers are augmented with bluetooth proximity sensors. The same sensors are also applied to safety equipment like helmets and gloves. In order to guarantee safety at all times, the tools are equipped with a 'dead man switch' that prevents the use if the proximity sensor do not detect the proper safety equipment in the close surrounding. The cards employed, the storyboard and cardboard are reported in Fig. 6.

The *scenario* used in the second idea is the same used in the first one, participation in public life, but the *persona* is different: a group of workers is addressed as target of the second idea.

The storyboard is composed of 6 illustrations. The second idea uses less cards than the first one, but they are well represented in the storyboard: workers, wearable gear, location change and proximity triggers are all present in the illustrations of the storyboard.

5.4 Perceived Ease of Use and Enjoyment

The results collected through the questionnaires are shown in Fig. 7. The following statements are reflected in the statistics:

- **S1**: Using the Tiles cards was fun;
- **S2**: The cards concepts were easy to understand;
- **S3**: The design process provided enough guidance to develop new ideas;
- **S4**: The information printed on the cards was useful;
- **S5**: The proposed users and contexts helped me framing the problem for a given person/society.

For the statements S1-S4, N = 57, for S5, N = 55. This discrepancy in the number of users is due to the fact that some of the participants didn't compile

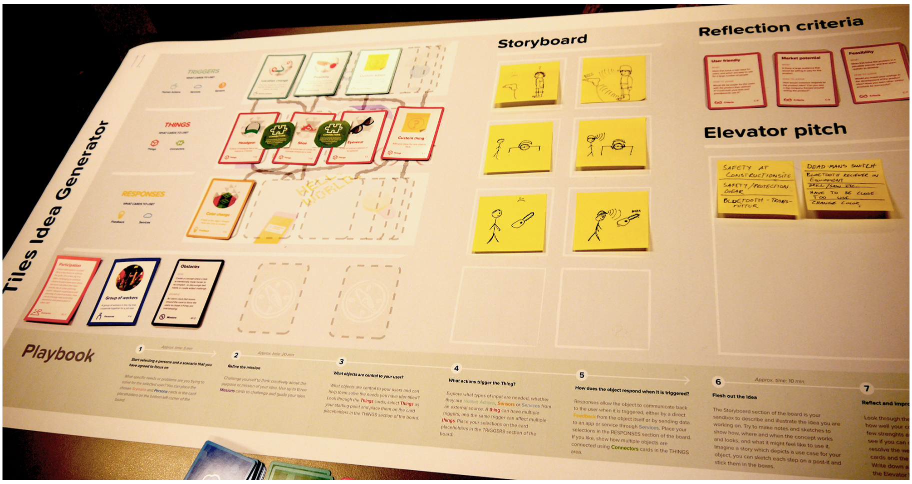


Figure 6. Cardboard and cards of the second idea.

the questionnaire or skipped part of it. The data show for all the statements a level of agreement between 60% and 85%. The disagreement is around 10% for S1 and S2, while it is much lower for S3, S4 and S5.

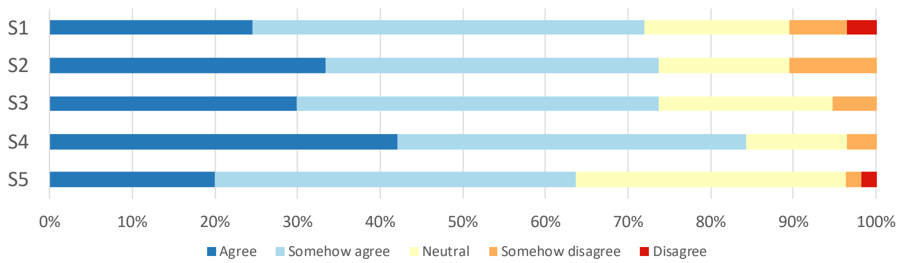


Figure 7. Results of the questionnaire statements.

6 Discussion

The goal of the smart city extension of the Tiles Ideation Toolkit is to support, motivate and empower non-expert users in ideation involving IoT and smart cities. This is envisioned as the first step toward the development of IoT collaborative applications that promote reflection and lifelong learning through increased citizens participation and awareness.

The smart city extension of Tiles successfully guided the users through the idea generation process: the final ideas matched most of the times the problem proposed in the *scenario* card, the target *persona* and made use of IoT technologies. The new *things* cards for smart cities and the *sensors* deck were often employed in the ideation process. This is an encouraging outcome, the adoption suggests user interest and the usefulness of the new cards. We cannot expect a usage close to 100% though, since the use of specific cards is dependent on the idea developed. For a few idea concepts, the users didn't need more cards than the ones provided by the original base decks.

The *scenario* and *persona* cards not only provided a design goal, but also modified the design process providing guidance and constraints, without limiting creativity. This emerged to be a critical factor, especially when the time at disposal for the workshop was limited. The two ideas analyzed make use of the same *scenario*, but different *personas*. In connection to this, two interesting outcomes emerged: (i) the two ideas have very few in common despite the fact that they share the same *scenario*, which suggests freedom in the creative process, (ii) the 'group of workers' was not interpreted exactly as intended, in fact the idea address a single worker, there are no elements that restrict the use only to a group and not to an individual. The division between individual and group based *persona* might be easy to grasp when reading and choosing the cards, but has proven to be challenging to maintain during the ideation process. A possible solution might be introducing further support or constraints during the workshop to remind the users about the difference, or as an alternative remove the distinction in the *persona* cards, leaving the users free to spontaneously generate an idea for an individual or a group.

In a couple of occasions we experimented running the workshop without *scenarios* and *personas*: the users were never able to come up with an idea or problem to solve, they spent all the time at their disposal browsing the cards without producing any result: a set of tools too generic hindered creativity and prevented ideation. Of course is important to be aware of the trade-off nature of these supporting factors: having too many constraints or guidance can inhibit the vision of the users. From the questionnaire comments and the observations, we registered mixed feedback regarding these two factors. However, no significant evidence indicating that the current level of constraints is inadequate emerged. Our evaluation didn't focus on this aspect in particular, but it may be of interest to test the workshop protocol using different levels of constraints, studying how they can affect the idea outcome. This can help to precisely adjust the protocol, aiming at the most effective level of guidance and constraints.

All the groups were able to generate an idea of an IoT application for smart cities. During the workshop they managed to produce the idea after only 40 minutes, a very limited amount of time, and pitch it after that. The ideas reported in Section 5.3 successfully retained the concepts of IoT as object augmentation and physical interaction between the target users and the augmented objects. The concept of ecology of interconnected smart objects is also present. The cards used on the cardboards are effectively scaffolding the applications, which are sketched in the storyboard and presented in detail during the pitch. The themes of the vast majority of the cards used are is still present at the end of the ideation process: for example the *persona* chosen, the *scenario* of interest and the *things* picked to be augmented were all mentioned when the idea was presented. Data from the questionnaire confirm that *persona* and *scenario* helped framing and contextualizing the process, providing additional support if compared to the original toolkit.

The storyboard demonstrated to be a useful step to bridge and refine the idea: it provided a simple instrument to quickly unfold the complete idea, facilitating the transition between the card based representation of the idea and the oral pitch. The storyboard also forced the participants to depict a use case for their concept, taking into account the temporal and cause-effect aspects of the flow of events composing the idea. These aspects were not explicitly enforced when the participants combined the cards in the first phases of the workshop.

In some occasions, a few of the cards placed on the cardboard were not present in the final idea pitch. This is anyway an acceptable twist, in the best case scenario the users didn't have enough time to go into all the details of the idea, but it is also possible that some concepts were simply abandoned during the process without removing the cards from the cardboard. Being the users non-expert in IoT, the real risk was that they could have fell back designing familiar application concepts, namely mobile apps or screen based applications. The toolkit and the design process effectively prevented this to happen at the end, although we have witnessed internal group discussions regarding the topics.

A positive outcome emerging from the user studies conducted is certainly the improved user independence during the workshop. It would have not been possible to run the workshop with more than 10 groups in parallel if the process would have required direct assistance by researchers and mentors. This is an important strength in the context of tool dissemination: the entire Tiles toolkit and the smart city extension are open source and freely available, the fact that they can also be used independently is a requirement for public adoption in learning and brainstorming contexts.

The low level of technical skills and knowledge required to use the toolkit is also supporting its diffusion. Most of the workshop users had no previous knowledge of IoT, smart cities or design methods, though they were perfectly able to complete the design process, sometimes with brilliant results in terms of originality and problem solving potential of the idea. The low barriers of adoption are particularly beneficial when promoting participatory design and co-creation for smart cities: extensive citizens involvement is still lacking in smart city applications [15] [9].

Based on our experience, schools and universities have demonstrated to be a valuable learning ecosystem where students can learn through the ideation of IoT applications. However, it also emerged that the same toolkit might yield significantly different experiences in different school environments. An effective toolkit is not enough, engagement and cooperation should also be supported by the teachers, which play a unique role connecting with the social circle of the students, facilitating the acceptance of an unfamiliar experience.

Although the goal of the smart city extension of Tiles is to stimulate creative thinking and ideation, the natural continuation of this process is the actual implementation of the idea, through prototyping and programming.

The extended Tiles toolkit can be a powerful tool for innovation, combining different cards can lead to exploring unprecedented solutions for smart cities. Creativity through serendipity is possible, from unexplored cards combinations innovative concepts emerge, which may constitute the core of the solution that users can refine through reflection and convergent-divergent thinking.

7 Conclusions

In this article we presented the smart city extension of the Tiles ideation toolkit, a card based toolkit supporting brainstorming of IoT applications. The smart city extension is composed of an additional set of cards divided into several decks and an updated workshop technique. Its goal is to facilitate the ideation of IoT applications for sustainability in the city.

The extended toolkit has been evaluated based on quantitative and qualitative data coming from a workshop with 60 users divided into 16 groups. Results showed that all the groups were able to generate an idea after a very limited amount of time. The participants found the extended toolkit fun, informative and helpful in providing guidance for idea generation.

We registered a good outcome in term of relevance of the ideas generated. The theme of IoT as object augmentation, the smart city elements and the ecologies of interconnected objects were all effectively used to scaffold the idea.

Future extensions of the toolkit can be oriented to tackle different scenarios of use like healthcare and governance, or other groups of *personas*. We plan to complement the ideation phase with prototyping and programming of the physical augmented objects. Thanks to low complexity programming paradigms to interact with sensors and actuators, end users can be facilitated in directly embedding the intended object behaviour into the electronics.

Acknowledgements

We would like to thank all the students and volunteers that participated in the evaluation workshops, and Dr. Simone Mora for his contribution and guidance during the workshops and his work on the Tiles toolkit.

References

1. Ashton, K.: That "internet of things" thing. RFiD Journal 22(7), 97–114 (2009)
2. Blomquist, A., Arvola, M.: Personas in action: Ethnography in an interaction design team. In: Proceedings of the Second Nordic Conference on Human-Computer Interaction. pp. 197–200 . ACM (2002)
3. Fleisch, E.: What is the Internet of things? An economic perspective. Economics, Management & Financial Markets 5(2) (2010)

4. Fowles, R.A.: Design methods in UK schools of architecture. Design Studies 1(1), 15–16 (1979)
5. Fu, Z., Lin, X.: Building the co-design and making platform to support participatory research and development for smart city. In: International Conference on Cross-Cultural Design. pp. 609–620. Springer (2014)
6. Gianni, F., Divitini, M.: Technology-enhanced Smart City Learning: A Systematic Mapping of the Literature. IxD&A 27, 28–43 (2016)
7. Greenbaum, J., Kyng, M.: Design at Work: Cooperative Design of Computer Systems. L. Erlbaum Associates Inc. (1992)
8. Hornecker, E.: Creative idea exploration within the structure of a guiding framework: The card brainstorming game. In: Proceedings of the Fourth International Conference on Tangible, Embedded, and Embodied Interaction. pp. 101–108. ACM (2010)
9. Klecha, L., Gianni, F.: Designing for Sustainable Urban Mobility Behaviour: A Systematic Review of the Literature. In: Conference on Smart Learning Ecosystems and Regional Development. pp. 137–149. Springer (2017)
10. Komninos, N.: Intelligent Cities: Innovation, Knowledge Systems, and Digital Spaces. Taylor & Francis (2002)
11. Koreshoff T.L., Robertson, T., Leong, T.W.: Internet of Things: A Review of Literature and Products. In: Proceedings of the 25th Australian Computer-Human Interaction Conference: Augmentation, Application, Innovation, Collaboration. pp. 335–344. OzCHI '13, ACM, Adelaide, Australia (2013)
12. Mechant, P., Stevens, I., Evens, T., Verdegem, P.: E–deliberation 2.0 for smart cities: A critical assessment of two'idea generation'cases. International Journal of Electronic Governance 5(1), 82–98 (2012)
13. Mora, S., Gianni, F., Divitini, M.: Tiles: A Card-based Ideation Toolkit for the Internet of Things. In: Proceedings of the 2017 Conference on Designing Interactive Systems. pp. 587–598. DIS '17, ACM, Edinburgh, United Kingdom (2017)
14. Nielsen, B.F., Juhasz-Nagy, E., Lindkvist, C., Wyckmans, A., Andresen, I., Baer, D.: Planning Instruments for Smart Energy Communities. PI-SEC Report 2.1. Tech. rep., NTNU, Sintef, Research Council of Norway (2017)
15. Reiersølmoen, M., Gianni, F., Divitini, M.: DELTA: Promoting Young People Participation in Urban Planning. In: Conference on Smart Learning Ecosystems and Regional Development. pp. 77–89. Springer (2017)
16. Schuurman, D., Baccarne, B., De Marez, L., Mechant, P.: Smart ideas for smart cities: Investigating crowdsourcing for generating and selecting ideas for ICT innovation in a city context. Journal of theoretical and applied electronic commerce research 7(3), 49–62 (2012)
17. Summit, W.G.: Smart Cities: Regional Perspective (2015)
18. Tian, R., Dierk, C., Myers, C., Paulos, E.: MyPart: Personal, Portable, Accurate, Airborne Particle Counting. In: Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems. pp. 1338–1348. ACM (2016)
19. Vermesan, O., Friess, P., Guillemin, P., Gusmeroli, S., Sundmaeker, H., Bassi, A., Jubert, I.S., Mazura, M., Harrison, M., Eisenhauer, M., others: Internet of things strategic research roadmap. Internet of Things-Global Technological and Societal Trends 1, 9–52 (2011)
20. Wagner, I., Basile, M., Ehrenstrasser, L., Maquil, V., Terrin, J.J., Wagner, M.: Supporting community engagement in the city: Urban planning in the MR-tent. In: Proceedings of the Fourth International Conference on Communities and Technologies. pp. 185–194. ACM (2009)

Paper 4

Tiles-Reflection: Designing for Reflective Learning and Change Behaviour in the Smart City

Francesco Gianni, Lisa Klecha, and Monica Divitini

In: The Interplay of Data, Technology, Place and People for Smart Learning (SLERD)

Is not included due to copyright
available at https://doi.org/10.1007/978-3-319-92022-1_7

Paper 5

Designing IoT Applications in Lower Secondary Schools

Anna Mavroudi, Monica Divitini, Francesco Gianni,
Simone Mora and Dag R. Kvittem

In: Proceedings of IEEE Global Engineering Education Conference (EDUCON)

DOI: https://doi.org/10.1109/EDUCON.2018.8363355

Is not included due to copyright
available at https://doi.org/10.1109/EDUCON.2018.8363355

Paper 6

RapIoT Toolkit: Rapid Prototyping of Collaborative Internet of Things Applications

Francesco Gianni, Simone Mora, and Monica Divitini

Journal of Future Generation Computer Systems

Future Generation Computer Systems 95 (2019) 867–879

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

RapIoT toolkit: Rapid prototyping of collaborative Internet of Things applications

Francesco Gianni *, Simone Mora, Monica Divitini

Norwegian University of Science and Technology, Department of Computer Science, Sem Sælandsvei 9, Trondheim, Norway

ARTICLE INFO

Article history:
Received 1 May 2017
Received in revised form 9 February 2018
Accepted 18 February 2018
Available online 27 February 2018

Keywords:
Internet of Things
IoT
Ubiquitous computing
Development
Toolkit
Smart cities

ABSTRACT

The Internet of Things holds huge promise in enhancing collaboration in multiple application domains. Bringing internet connectivity to everyday objects and environments promotes ubiquitous access to information and integration with third-party systems. Further, connected "things" can be used as physical interfaces to enable users to cooperate, leveraging multiple devices via parallel and distributed actions. Yet creating prototypes of IoT systems is a complex task for developers non-expert in IoT, as it requires dealing with multi-layered hardware and software infrastructures. We introduce RapIoT, a software toolkit that facilitates the prototyping of IoT systems by providing an integrated set of technologies. Our solution abstracts low-level details and communication protocols, allowing developers non-expert in IoT to focus on application logic, facilitating rapid prototyping. RapIoT supports the development of collaborative applications by enabling the definition of high-level data type primitives and allowing interactions spread among multiple smart objects. RapIoT primitives act as a loosely coupled interface between generic IoT devices and applications, simplifying the development of systems that make use of an ecology of devices distributed to multiple users and environments. We illustrate the potential of our toolkit by presenting the development process of an IoT application ideated during a workshop with non-expert developers and addressing real-world challenges affecting smart cities. We conclude by discussing the strength and limitations of our platform, highlighting further possible uses for collaborative applications.

1. Introduction

The Internet of Things (IoT) holds huge promise in enhancing computer-supported collaboration in several application domains. By enabling seamless interconnection of people, computers, everyday objects, and environments, it promotes collaboration off the screen in our everyday routines. Additionally, increasing the amount and quality of information captured by connected objects might ultimately improve collaboration among people using those objects [1]. The technological matrix of Computer Supported Cooperative Work (CSCW) is evolving to facilitate context-aware computing, mobile communication, and interaction. Support for this paradigm shift also comes from the already established collaborative potentials and implications of the IoT [2].

Research has shown how IoT systems can leverage connected objects in collaborative applications, for example, to support patient/physician dialog in chronic disease treatments [3], to foster social communication among friends and relatives [4], to enhance collaboration in crisis management [5], and to support citizens' participation in public administration [6].

Since the term "Internet of Things" was coined in 1999 by technologist Kevin Ashton [7], research has mainly focused on developing machine-centric infrastructures to enable connected things to exchange information over the internet. Wireless sensor networks (WSN), machine-to-machine (M2M) communication, and technologies connected to design, deployment, and operation of WSN are some of the most common topics of interest.

Few works [8,1] have investigated how the IoT can enable collaboration and how HCI theory could drive the development of IoT collaborative applications. Likewise, only a few works have investigated collaborative IoT application authoring [9] and how to involve users in design activities [10,11].

We define a collaborative IoT application as a technological application where users are engaged in a joint effort, having the ability and flexibility to align their interactions through the support provided by ecologies of interconnected things.

The CSCW agenda has become relevant for these activities, having expanded out of the boundaries of the work environment, and diversified into new areas of human activity. The social organization of these activities, as well as the intertwining of social science with computer science to explore, inform, and propel technological research, is of crucial importance to the continued development of CSCW [12]. Further, connected and interactive objects have been

* Corresponding author.
E-mail addresses: francesco.gianni@ntnu.no (F. Gianni), simone.mora@ntnu.no (S. Mora), monica.divitini@ntnu.no (M. Divitini).

https://doi.org/10.1016/j.future.2018.02.030

employed in CSCW applications for long time; for example as awareness and coordination devices [13].

With RapIoT, we concentrate on supporting interaction in the physical world in connection with the digital domain. Starting from a list of design goals, grouped in infrastructure, support for developers, and support for collaborative applications, we propose an architecture oriented to support IoT applications that make use of tangible interaction through smart objects in the physical world. We foresee "things" and smart objects as enabling artifacts for shared, collective, and collaborative activities [14]. This concept has its roots in Greenberg's conceptualization of *physical collaborative interfaces* [15] as devices situated in the physical world and designed for collaborative use. These devices may retain the appearance of everyday objects but are able to collect data, visualize information, provide feedback, and sense user interaction and manipulation.

This is a substantial and novel paradigm shift for IoT applications, since lack of mobility is a typical limitation of common IoT devices [16].

The target end user of a RapIoT application needs only to be able to physically interact with the augmented object, which does not require any particular skill, since the object retains its original affordances.

We define RapIoT developers as "non-experts" in IoT. They differ from professionals in the fact that they do not have any skill in electronics, networking protocols, or the assembling and configuration of IoT devices. They do have some programming proficiency and are able to code using standard constructs of high-level programming languages such as JavaScript, Python, etc.

Makers, designers, and students are examples of non-expert developers who can be part of a participatory design-oriented strategy to allow a wide population to take advantage of the potential that IoT technologies offer for collaborative applications.

We foresee that their involvement in design and programming for the IoT will result from a lowering of the threshold of skills required to build prototypes. Although a number of tools are available to support IoT development, those tools often (i) do not offer integrated support to multiple architectural layers, (ii) require pre-existing knowledge in hardware development or embedded programming, and are thus not suitable for non-experts in IoT, and (iii) are often bound to specific hardware and vendor-locked technologies. This results in a steep learning curve for the tools and large time for integration, obstructing the ability and speed with which developers may explore design choices by iterating on the implementation of functional prototypes.

Rapid prototyping is an important development process when creating innovative IoT applications. Ideas can be quickly tested and refined, keeping costs low. Through rapid prototyping, we aim to encourage and engage non-expert developers in exploring the vast solution space offered by IoT technologies. However, prototyping IoT systems is challenging because doing so requires dealing with a heterogeneous mix of hardware and software components arranged in a multi-layer architecture. Lowering the entry barriers and facilitating adoption are steps needed to achieve participation in brainstorming and other collective activities [17]. Toolkits for IoT address these issues: They provide an integrated set of technologies and practices to simplify and scaffold prototyping.

A popular architectural pattern for IoT toolkits consists of three layers [18]:

- an *embedded layer*, implemented as a physical object augmented with sensors, actuators, and short-range wireless connectivity to provide sensing and user interface capabilities;
- a *gateway layer*, implemented as a device such as a smartphone or WiFi router, to provide connectivity to the embedded layer, enabling ubiquitous access to information;
- a *server layer*, implemented as a cloud service, which enables data storage and integration with third-party services.

RapIoT provides support for all three layers of this type of architecture. RapIoT does not explicitly support a specific application domain, acting as an enabling technology allowing non-experts to develop collaborative applications. From this perspective, RapIoT enables the definition, implementation, and manipulation of high-level *data type primitives*. RapIoT primitives allow developers to abstract out low-level implementation details and provide a loosely coupled interface between different architectural layers, allowing IoT devices to serve different applications without the need for firmware reprogramming and thus offering a platform as a service. They introduce a simple shared construct that traverses the three layers of the architecture, providing continuity and facilitating rapid prototyping and deployment of IoT applications.

We envision the emerging domain of collaborative IoT applications for smart cities as a possible application context, where the city is not merely a group of persons but a vibrant ecosystem of communities. Through collaborative practices involving multiple artifacts, citizens build awareness and enable lifelong learning [19]. We concentrate on this domain, which is particularly timely and fitting, given the stagnation of technological advancements in applications for smart cities [20] and the potential impact on society, supporting improvements of all key factors contributing to regional competitiveness: mobility, environment, people, quality of life, and governance [21].

In Section 2, we provide an analysis of existing IoT frameworks and toolkits. In Section 3, we summarize the characteristics of an IoT toolkit that can support non-experts in developing collaborative applications. We then present *RapIoT*, an integrated set of tools to support rapid prototyping of IoT applications, previously introduced in [22]. The RapIoT approach is then described in detail, in relation to IoT applications developed in the smart city domain. We discuss the strengths and weaknesses of our approach, and we conclude the paper by highlighting future works.

2. Related work

Several works have provided tools to facilitate the development of IoT systems. Aside from relying on standard protocols and APIs that allow mutual integration, each tool often focuses on supporting a specific architectural layer. In the remainder of this section, we survey development toolkits that can be used for IoT prototyping.

2.1. Embedded layer

Modkit [23] extends the Arduino [24] platform providing a block-based visual programming language based on the Scratch project [25], further expanding Arduino target users to non-professional developers such as kids, designers, and artists. Focused on developing interfaces based on simple input/output feedbacks, Bloctopus [26] provides a platform based on modules with sensor-actuator couplings and a hybrid visual and textual programming language. Micro:bit [27] is a small electronic board equipped with a microcontroller, a low-fidelity display, and a few sensors. Micro:bit can be programmed with high-level programming languages and has been used extensively in schools.

2.2. Gateway layer

Developing or deploying gateways to provide internet connectivity to resource-constrained embedded devices is particularly limiting for non-experts, as it requires pre-existing knowledge of low-level technologies such as transport protocols and wireless networks.

Fabryq [28] simplifies the development and deployment of internet gateways for Bluetooth low-energy (BLE) devices by abstracting the complexity of dealing with multiple languages and networking aspects. Rather than invoking BLE commands on each local device, the platform provides a proxy to access multiple devices via a centralized API.

Zhu et al. [29] have addressed the development of a gateway for ZigBee[1] wireless devices. IoT devices can be controlled and accessed remotely, and the gateway handles conversion between different data protocols.

Commercial IoT gateways such as Libelium's Meshlium[2] and Multitech's MultiConnect Conduit[3] are standalone fixed devices which provide a bridge between WSNs and the cloud. Conduit offers BLE connectivity for IoT devices, while Meshlium relies only on the ZigBee protocol to communicate with the sensors. On the cloud side, they both offer WiFi and 3G/4G connection; MQTT protocol[4] is also supported. Conduit's onboard software can be developed, depending on the specific model, using either the Node-RED visual programming language or the mLinux development environment, which allows coding in C++, C#, Python, and Java, among others. Meshlium allows only local data storage or the transfer of sensor readings to a list of supported cloud services such as IBM Bluemix, Microsoft Azure IoT Hub, and Amazon IoT.

2.3. Server layer

The server layer is the core element that manages IoT devices connected via multiple gateways and interacts with third-party web services such as data providers or social networks.

The framework *PatRICIA* [30] leverages a programming model and a cloud-based execution environment to reduce complexity and support scalable development of IoT applications. Similarly, the framework developed by Khodadadi et al. [31] focuses on connecting data sources by managing querying and filtering of data and facilitating sharing with third-party platforms. Their work takes into account data gathering from multiple sources such as sensor networks and other web applications (blogs, social media, databases). Users are provided with an API to configure data sources and to trigger actions within stand alone applications. Kovatsch et al. [32] describe a similar higher-level architecture. They address the need for an API for connected devices for pushing and retrieving data.

IFTTT is an online platform to connect event conditions, called "triggers", generated by a device or online service, to "actions" associated with other devices or services. IFTTT is not exclusively oriented to the IoT but supports a number of physical smart devices that can be used to trigger events or to perform actions in response to a triggered event.

SpaceBrew[5] is a software toolkit to connect interactive things to each other, which can be defined as publishers or subscribers. The data is exchanged as Boolean, numeric, or string values. Data processing is handled by the interactive things themselves in addition to sensing or actuation.

Paraimpu [33] allows developers to connect sensors and actuators through a centralized RESTful service. Data is exchanged in several formats, among which are numeric, text, JSON, and XML. A simple programmable layer between sensor and actuator allows developers to specify filters and conditional logic. The programming language to code the logic is dependent on the data type used by the sensor, for example, RegEx, JavaScript, or XPath. Internet services like Twitter can act as a sensor and be connected to actuators. Arduino can be used as an actuator, but a specific sketch should be generated and downloaded to embed the logic and to handle the output pins.

Node-RED is a visual data flow programming language (VDFPL) which also targets IoT scenarios. It allows developers to create flows connecting self-contained blocks which are treated as black boxes, following the principles driving the VDFPL paradigm.

Shiftr.io is an online MQTT broker as a service. It allows interconnection of MQTT clients and online message flow visualization. Data and MQTT connected services sharing is publicly encouraged by the platform's design. Shiftr.io uses the same MQTT messaging protocol adopted in RapIoT. The service targets the IoT facilitating the interconnection of MQTT clients, which can run on different types of hardware devices.

Amazon AWS offers two IoT oriented services: AWS Greengrass and AWS IoT Platform. Both provide secure messaging among devices, connection to the AWS cloud and an SDK to code the application. Greengrass targets the gateway layer, while IoT Platform addresses the cloud.

WoTkit [9] is a toolkit for IoT *mashups*: web applications that blend data and services available on the web with physical data sources such as IoT devices. Data is combined and visualized through a dashboard, accessible using a browser. WotKit focuses less on the integration of applications, rather providing basic built-in visuals and processing components [9].

2.4. The RapIoT position

Our system takes advantage of the Arduino platform but differs from the approach used by Modkit, Micro:bit, and Bloctopus since we do not try to include any application logic in the embedded layer. In RapIoT, the implementation of the embedded layer simply provides a domain-specific language (DSL) as an Arduino library whose only purpose is to facilitate the definition and coding of input/output primitives. The approach used in Fabryq [28] requires pre-existing knowledge about the BLE protocol, so the toolkit is not suitable for the skill level of non-expert developers. The gateway solution by Zhu et al. [29] implies that only the parent node is connected to the network; child nodes are not directly accessible from the application environment, hindering multi-object dynamics in the application logic. *Conduit* gateway is a powerful device, but its limitations mainly consist of the price being around ten times the price of a smartphone with the same connectivity, processing power and of the device not being designed to be power efficient or portable. Both *Conduit* and *Meshlium* are devices clearly oriented to WSNs deployed in a fixed environment. They do not fit in a scenario where deployment flexibility is a requirement: To support a new sensor topology, the application on the gateway needs to be updated and redeployed. This process requires an external computer and programming skills in the languages supported. The proposed approach and implementation of *RapMobile* is suited to solve many of the limitations conventional IoT gateways present. Zachariah et al. [18] describe in detail the shortcomings of current IoT gateways. Their envisioned solution presents many points in common with *RapMobile*, including (i) disentangling of network connectivity, in-network processing, and user interface functions; (ii) leveraging of BLE connectivity for IoT devices and a mobile application as a gateway; (iii) providing application-agnostic connectivity; (iv) using a single mobile application to connect heterogeneous BLE devices in an opportunistic way; and (v) moving the power burden of WiFi/3G/4G network protocols from the IoT devices to the gateway. The framework *PatRICIA* [30] focuses on providing sensor management in a cloud environment and storing data received from connected devices, neglecting interaction with other third-party solutions. They also neglect the management of

---

1 http://www.zigbee.org.
2 http://www.libelium.com/products/meshlium/wsn.
3 https://www.multitech.com/brands/multiconnect-conduit.
4 http://mqtt.org.
5 http://docs.spacebrew.cc/about.

Table 1
Architecture layers covered and non-expert developers support of related works and RapIoT.

	Modkit	Bloctopus	Fabryq	Zhu et al. [29]	Meshlium	Conduit	PatRICIA	Khodadadi et al. [31]	Kovatsch et al. [32]	IFTTT	SpaceBrew	Paraimpu	Node-RED	Shiftr.io	WoTkit	AWS	RapIoT
Embedded	•	•	•								•						•
Gateway			•	•	•	•										•	•
Cloud			•				•	•	•	•	•	•	•	•	•	•	•
Non-experts	•	•								•			•	•		•	•

connected devices through an API and rather focus on reading and combining data from different sources. Each device needs to be directly connected to the cloud through the MQTT protocol, which prevents the inclusion of mobile and low-powered IoT devices. The solution proposed by Kovatsch et al. [32] enables devices to publish data to third-party servers but does not support bi-directional exchange of events in real-time. The restrictions of IFTTT lie in its supported services, devices, triggers, and actions, which are limited to those offered by the platform and not extensible by end users. Users are then constrained to mixing and matching triggers and events already implemented. SpaceBrew and Paraimpu do not facilitate deployment of applications; the user has to find a way to connect the things, sensors, and actuators to the web. They also lack a unified structure for the messaging protocol and data format. In Paraimpu, the programming logic is also dependent on the data format, requiring programming skills in different languages even for simple scenarios. Both platforms allow only very simple logic constructs in the applications, and SpaceBrew is also constrained by the data type used: For example, it is not possible to connect a publisher of Boolean values to a subscriber that handles strings. In SpaceBrew, computation does not happen in the cloud but on the devices, which by definition are usually quite limited in processing power and are often running on batteries. The Node-RED platform is based on an interesting programming approach which can possibly complement the RapIoT platform. Node-RED supports the programming phase but is not a full-stack toolkit intended to handle and scaffold deployment, hardware device programming, and data flow from BLE-enabled sensors and actuators. Although Shiftr.io provides some advanced MQTT functionalities such as graph-based live visualization of messages and data sharing, the core function remains to serve as an online MQTT broker. Limitations are imposed on the supported Arduino hardware, which should have onboard WiFi/Ethernet connectivity, restricting the choice to few boards such as Arduino Yún or others that require cable-based connectivity.[6] RapIoT already integrates an online MQTT broker supporting the message flow from the hardware devices to the programming environment. Shiftr.io does not offer an integrated full-stack platform, presenting similar limitations as the ones discussed for Node-RED. The Amazon AWS services for IoT allow developers to connect IoT devices to the Amazon cloud. However, no low-power BLE hardware is supported, and the devices are required to have support for WiFi/Ethernet connectivity and run a Linux OS or a software stack such as Python, Node.js, or Java, which is not usually supported by low-power microcontrollers. No support is provided for a gateway able to bridge BLE connections from IoT devices to the cloud. Compared to WoTkit [9], RapIoT focuses more on interaction in the physical world. No widget or computer-based visualizations are supported since the architecture of RapIoT is oriented to tangible interaction through smart objects in the physical world.

6 https://github.com/256dpi/arduino-mqtt.

Table 2
Hardware and infrastructure.

A1	**Support both novice and expert developers** — provide basic, simple-to-use functionalities without hindering expert users in building complex systems.
A2	**Decouple infrastructure from application** — provide the IoT infrastructure as a service to applications. In this way the infrastructure (IoT devices, gateways, and servers) is completely decoupled and can be reused across different applications with no or little changes.
A3	**Hide hardware complexities** — provide high-level representations of low-level embedded hardware complexities.
A4	**Hide networking details** — spare developers from implementing connection and data transfer protocols.
A5	**Support for generic embedded devices** — enable the development of applications that make use of a wide range of IoT devices, no matter of manufacturer.
A6	**Support for multiple embedded devices** — enable the development of IoT systems that make use of multiple devices which collaborate as a structured ecology.

2.5. Summary of differences with related works

We reviewed toolkits that can be used to support the development of the embedded, gateway, and server layers of an IoT infrastructure and summarized the results in Table 1. Current solutions often support only a subset of the architectural layers of a common IoT toolkit. The knowledge required to use each tool also varies according to the level of abstraction it provides and the complexity of the applications that can be achieved. With RapIoT, we chose to relax some of the constraints typically found in single-layer toolkits – namely device-specific optimizations or fine-tuned trade-offs among energy efficiency, accuracy, and latency – to gain a better support for non-expert developers and rapid prototyping.

3. RapIoT fundamentals

3.1. RapIoT architecture: design goals

RapIoT aims at providing holistic support to non-experts developing collaborative IoT applications. The following architectural design goals constitute the foundation of our platform; we grouped them into (i) hardware and infrastructure guidelines (Table 2), (ii) directions to support non-expert developers (Table 3), and (iii) characteristics supporting collaborative applications (Table 4).

With these goals, we promote hands-on collaboration based on the shared physical experience of a small community. At the same time, we also support asynchronous coordination and information sharing via connected services.

3.2. Input/Output primitives

One of the crucial features of RapIoT is the concept and implementation of high-level input/output primitives. We envisioned a

Table 3
Support for non-expert developers.

B1	**Employ technology close to the user** — use of mainstream solutions and technologies that are easily accessible and widespread.
B2	**Provide an efficient workflow** — minimize the time needed to deploy a first prototype of an IoT application; this includes using programming languages which allow for quick evaluation of the application code.
B3	**Have a low cost** — adopt low-cost hardware and software solutions which are free to use and preferably open source.
B4	**Have a generic architecture** — provide a structure that is adaptable and extensible to different problem domains.
B5	**Empower community support** — facilitate cooperative work and reuse of knowledge. All the points above allow for community-based sharing of knowledge and support.

Table 4
Support for collaborative applications.

C1	**Support for coordination of interdependent activities across space** — which is one of the problems faced by actors engaged in cooperative work *in the wild* [34].
C2	**Integration with third-party services** — provide hooks for web standards and cloud computing, which are base technologies for IoT systems [35].
C3	**Support for tangible interaction, physical user interfaces, and smart objects** — use of physical affordances to interact with computer systems, which have been proved effective in supporting collaboration [36, p. 97]. The IoT can leverage physical and embodied interaction approaches to interact with the "things".
C3	**Interaction spread among multiple things** — support a user experience distributed on an ecology of devices, providing more opportunities for collaboration via distributed actions performed by users on multiple interfaces.

developer-friendly construct that could be easily grasped by non-experts while supporting data exchange in collaborative multi-object IoT applications. Making primitives human readable facilitates development and debugging as opposed to dealing with raw sensor data. A RapIoT *input primitive* is discrete information sensed by an IoT device, for example, a data-point captured by a sensor or a manipulation performed via a user interface. An *output primitive* is an action that can be performed by the IoT device via output components such as actuators or displays, for example, a motor spinning or an LED (light-emitting diode) blinking (Fig. 1). Primitives act as a loosely coupled interface between embedded devices and the application logic. Each primitive encapsulates a data type plus up to two optional parameters as payload. An example of an input primitive is "AirQuality *(primitive name)*, city center *(parameter 1)*, low *(parameter 2)*" in case of an air quality sensor device or "Knocked, twice" in case of a smart home equipped with an accelerometer device on the front door. Otherwise "Vibrate, long" represents an output primitive that issues a vibrate command to a necklace equipped with a haptic motor device.

The role of primitives is twofold. They allow an event-driven approach to programming, providing at the same time simple constructs to describe the data exchanged between embedded devices and applications. Furthermore, they allow non-expert developers to think in terms of high-level abstractions without dealing with hardware complexities, e.g., "shake, clockwise rotation, free fall" for physical manipulations detected by accelerometer data. The definition, implementation, and registration of primitives is performed by programming the firmware of an Arduino-compatible device, and the primitives are then available to the toolkit. When coding the firmware, it is possible to deal with low-level hardware details, for example, accelerometer or GPS sensors as well as motor or display actuators. Primitives not only support simple input/output operations, they can also encapsulate more complex behaviors to support the development of physical interfaces, as

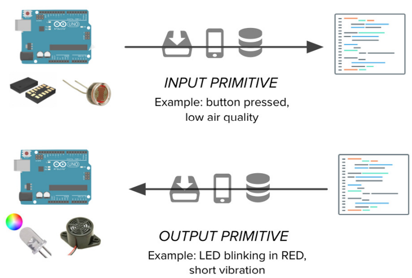


Fig. 1. Structure of input and output primitives.

illustrated in [11]. An example of HCI primitive introduced in [11] is the *"proximity"* input primitive. The primitive does not encapsulate any sensor data from the surrounding environment but is triggered when one or more IoT devices are moved close to one another. It is available to be used by devices that have the on-board hardware to support the functionality (i.e., RFID antennas and tags to sense one another). Primitives completely rely on the operations supported by the hardware, both in terms of input and output capabilities. They are bound to the hardware device and its sensing and actuation means. The gateway and server layer do not embed any specific list of primitives; rather, these two architectural layers transparently allow the programmer to work with any primitive supported by the hardware when coding the application.

3.3. Architecture

We now present the architecture of RapIoT, describing the requirements for the hardware, supported devices, software features and developer interaction. RapIoT follows the three-layer architecture described in Section 1 and represented in Fig. 2. The implementation of each software stack is discussed in Section 6.

Server layer

Rapcloud: This consists of an online IDE,[7] a JavaScript library, and a web-based configuration utility, as seen in Fig. 3. The online IDE is based on the Cloud9[8] platform, which combines a code editor and a back-end server workspace based on Ubuntu[9] Linux. The whole platform is provided as SaaS (software as a service) through the browser; developers do not need to install any software on their own computers, as the application code developed is automatically saved and ready to be executed directly on the server. To get started with the application coding, as a first step, the developer gets access to the web-based configuration utility and signs up, choosing a username and password. She can then create the skeleton of her first IoT application, picking a name and adding as many *Virtual Devices* as needed. *Virtual Devices* are placeholders used as IoT device handlers. They are available in the form of JavaScript objects when writing the application code. As an example, for a "smart shower" application, two *Virtual Devices* can be named "shower handle" and "shower light". To add a *Virtual Device*, the developer needs only to specify its name; no other

7 Integrated Development Environment.
8 https://c9.io.
9 https://www.ubuntu.com.

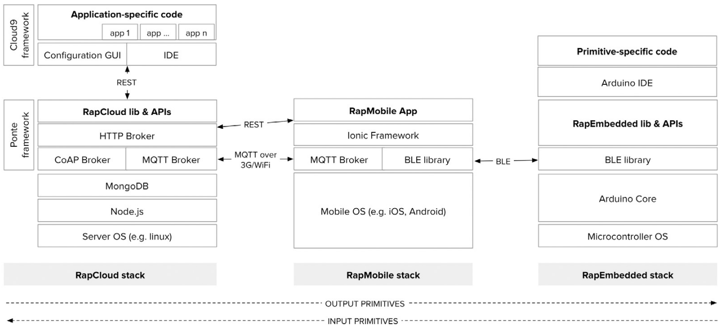


Fig. 2. RapCloud infrastructure implementation.

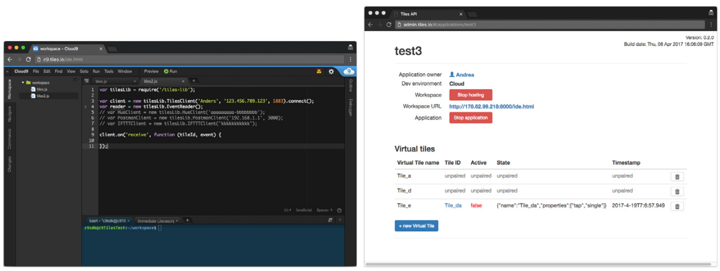


Fig. 3. RapCloud infrastructure: the Cloud9 online IDE on the left and the web-based configuration utility on the right.

information is needed. The developer can then launch the Cloud9 IDE, where a precompiled JavaScript source file is made available to be extended with custom application code. The precompiled source file includes the JavaScript objects of the *Virtual Devices* previously defined, ready to be employed by the developer when coding the application logic and handling the primitives.

Gateway layer

RapMobile: This is a cross-platform mobile app for Android and iOS devices that acts as an internet gateway and allows developers discover and configure IoT devices. The app at first connects to the *RapCloud* server, and a username and password must be entered to identify the user. The end user is then presented with the list of applications that the developer previously created in *RapCloud*, fetched directly from the server. Tapping on the application that needs to be launched, the user is faced with the list of *Virtual Devices* for that application, previously defined in the *RapCloud* environment. Following on our "smart shower" example, the "shower handle" and "shower light" entries will be visualized. Tapping on each *Virtual Device*, the user can associate it to a BLE IoT device available in proximity to the smartphone. Since there might be several BLE devices in the vicinity, the user is presented with a list to choose from, containing the BLE advertised names. Tapping on one of the advertised names associates the IoT device to the *Virtual Device*. The association is stored locally on the *RapMobile* app and is remembered until the user deletes it. Once all the *Virtual Devices* have been associated to physical devices, the IoT application (i.e., the "smart shower" application) can be started directly from the mobile app. Whenever the application is running, the phone can be set in standby mode, but it should remain within a 10 meter reach of the hardware devices to ensure reliable data transfer. We call this process of association between virtual and physical devices *application appropriation*. The GUI supporting appropriation and execution of the application is shown in Fig. 4.

The *RapMobile* app transparently bridges the BLE and MQTT protocols. When a primitive is received from a BLE device, the internal components of the app re-route the complete primitive packet through the MQTT connection with *RapCloud*. The *RapMobile* gateway includes all the needed information to correctly route the primitives: For every input primitive received, the sender IoT device is known, and the associated *Virtual Device* belongs to an application running on *RapCloud*, to which the MQTT packet is forwarded. The same packet routing steps apply in inverse order when an output primitive is forwarded from *RapCloud* to the IoT device.

Embedded layer

RapEmbedded: This consists of an Arduino library to support the definition and implementation of input and output primitives on electronic Arduino boards and microcontrollers. Arduino is a popular prototyping platform which includes both a microcontroller board to which sensors and actuators can be wired and a software library created to simplify writing code without limiting flexibility [24]. The Arduino library spares developers from learning microcontroller-specific instructions or electronic principles. Device-specific optimizations are handled by the deployment toolchain, which compiles the Arduino code into a microcontroller-optimized binary.

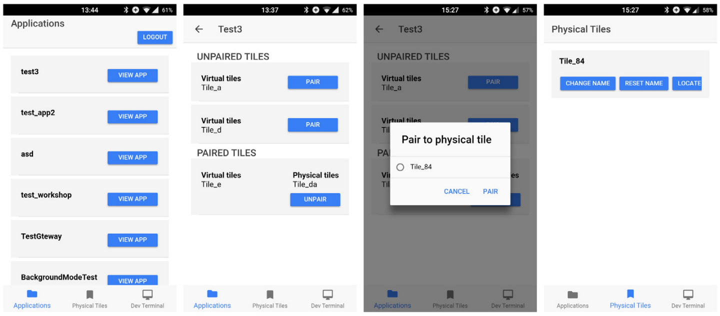


Fig. 4. RapMobile application.

3.4. Hardware requirements

The server-side software runs completely on Linux. A low-end Linux server or a small Linux virtual machine will suffice to cover the requirements. The requirements for the device running the mobile application, typically a smartphone, are limited to providing BLE and WiFi or cellular connectivity. The BLE interface is used to connect one or more Arduino-enabled boards, while a WiFi or cellular network provides TCP/IP connectivity to the cloud. The lowest architectural layer comprises BLE-enabled, Arduino-compatible boards and microcontrollers such as RFDUINO[10] and Simblee[11] boards. RFDUINO is able to run the *RapEmbedded* firmware without introducing any bottleneck in terms of CPU speed, flash, or RAM size. The microcontroller is based on a 16 MHz ARM Cortex-M0 CPU, 128 kB of flash memory and 8 kB of RAM. There are no other technical requirements for the embedded layer; the board can be equipped with additional I/O ports, extra connectivity, or sensors/actuators. Thanks to the open-source nature of Arduino, several independent producers were able to add Arduino compatibility to their products, allowing non-expert developers to choose from several royalty-free, affordable BLE-enabled microcontrollers and boards while at the same time taking advantage of the support provided by the growing Arduino community. Expert developers can still choose to build their own hardware, assembling the electronic to create a smart augmented object, while non-experts can simply buy a pre-assembled Arduino board, already equipped with sensors and actuators. Arduino boards are typically low power, small sized and can run on small batteries. RapIoT builds on Arduino's strengths and extends a similar approach to the IoT world. Developers interested in building applications are offered a set of primitives tailored and specific to the affordances of the IoT hardware in use, but at the same time they share a common semantic structure and are used in the same way when coding the application logic. Another point in common is the abstraction of vendor-specific programming mechanisms: Like the Arduino user, who is not required to know the type and producer of the microcontroller, RapIoT developers are not required to know any hardware- or software-related details of the IoT device. The non-expert developer need only to be aware of the set of primitives defined and available to be used for application development. The *RapEmbedded* software layer might be completely transparent to non-expert developers, who can buy a pre-programmed Arduino board embedded with the library and a firmware to handle input/output primitives. More skilled developers can decide to upgrade the firmware with a community-developed version, which can, for example, implement more or different primitives for the hardware in use. While no programming is needed to upgrade the firmware, expert developers are free to implement new primitives using the *RapEmbedded* Arduino library and flashing the new firmware on the board afterward. The *RapEmbedded* library provides functions to (i) register primitive definitions according to the name of the primitive, type (input or output), and name of (up to two) optional parameters and (ii) code conditions under which primitives are triggered, in case of input primitives, or consumed, as for output primitives.

10 http://rfduino.com.
11 http://simblee.com.

4. Creating RapIoT applications

In this section we illustrate how RapIoT can be employed to support the prototyping phase of an IoT application. The list of required steps and their relation with RapIoT components is reported in Fig. 5. We use an IoT application for smart cities as a running example.

RapIoT-supported application development and deployment is a four-step process. The first and second steps entail application development by developers, while the last two involve application appropriation by end users.

Step 1: *Device development* — This involves, in order of complexity for non-experts, either of these options: (i) building a hardware prototype of an IoT device using electronic components on a BLE-enabled, Arduino-compatible board using the *RapEmbedded* Arduino library to implement and register input/output primitives and flashing the firmware on the Arduino board; (ii) purchasing a complete IoT device or smart object based on an Arduino BLE microcontroller, coding the primitives as explained in the previous option or flashing a pre-made firmware embedding a set of primitives for the specific device; and (iii) purchasing a ready-to-use, RapIoT-compatible IoT device or smart object that mounts out of the box a firmware with coded primitives.

Step 2: *Application development* — This entails creating an application through the web-based configuration utility and coding the application logic by using the online IDE provided by *RapCloud*. Input and output primitives are here employed as programming constructs.

Step 3: *Application appropriation* — This involves selecting and starting from the *RapMobile* app an application previously developed on *RapCloud* and performing the wireless discovery of the BLE-enabled devices built in Step 1, as previously described in Section 3.3.

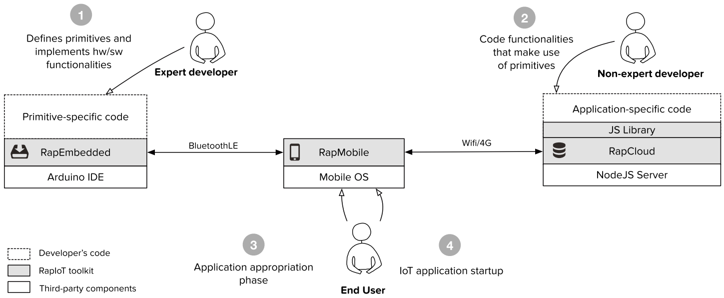


Fig. 5. The RapIoT toolkit, development and deployment process.

Step 4: *Application startup* — This entails tapping the "start application" button on the *RapMobile* app. The IoT application source code hosted on *RapCloud* is then executed.

These steps do not require advanced skills in hardware, electronics, or network protocols, only a general knowledge in coding using high-level programming languages. This matches our definition of non-expert developers reported in Section 1.

To describe the development process of RapIoT applications, we introduce as a running example the development of an application idea generated using the Tiles Ideation Toolkit for IoT [37]. The idea was produced during a workshop which involved computer science university students and consists of a *Smart Shower*, an augmented shower to promote learning of sustainable behaviors in children. The application targets children and parents, requiring a collaborative approach at a family level. The *Smart Shower* makes use of several connected IoT devices that provide feedback to the user, connect with online services, and sense the user interaction aimed at controlling the water flow. The system makes use of an IoT device that connects to the shower water handle. When the handle is operated by the child, the tilt is sensed by the IoT device, which can then infer water temperature and flow. This information is sent to the *RapCloud* server, which computes the energy consumption based on the temperature and quantity of the water used. The application logic is then configured to trigger different types of feedback based on water and energy thresholds reached. A second IoT device, in the form of an LED array governed by a BLE-enabled Arduino microcontroller, is used to provide shared ambient feedback that are visible to individual users or to the family, for example signaling that a weekly family consumption threshold has been reached, a common goal has been achieved, or visualizing triggers to collaborative reflection based on individual consumption compared to the rest of the household. When the *RapCloud* application computes the reach of a threshold, it triggers an output primitive addressed to the LED array, which changes color from green to yellow or from yellow to red when too much water is being used. The *RapCloud* application collects data about water and energy usage, which is then shared using an online spreadsheet or charting service. Based on the data, parents can reward their children for sustainable behaviors. Families can set group goals for weekly or monthly consumption and check their progress day by day, increasing community awareness and improving collaboration to reach the shared objective. In the following, we describe the applications development and deployment process.

4.1. Device development

Device development includes hardware and firmware development. Hardware development involves plugging together electronic components using an Arduino-compatible BLE board (Fig. 6).

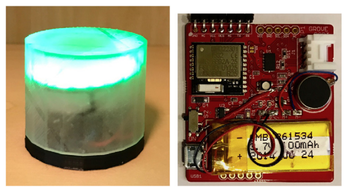


Fig. 6. LED array light and Tiles Square module hardware devices. The Tiles Square measures approximately 4 × 4 cm.

Firmware development requires writing Arduino code that controls the hardware and handles input and output of the primitives. According to our example, the electronic for the Smart Shower uses of a Tiles Square module (an Arduino-compatible BLE board) [11] and an LED array (Fig. 6). The Tiles Square module embeds in a single compact package an RFDUINO microcontroller, an accelerometer, an RGB LED light, and haptic feedback vibration.

After having installed the *RapEmbedded* library in the Arduino IDE, the developer defines the input primitive *Orientation* and the output primitive *LightColor*. The *Orientation* primitive models the 3D position of the shower handle; it is triggered by sensor readings continuously provided by the accelerometer on the Tiles Square and has a *Position* parameter that reports the angular orientation on X, Y, Z axes. The *LightColor* output primitive defines the *color* parameter which can assume "green", "yellow", and "red" states and causes the LED array to light up in different colors.

```
RIOTe.regPrimitive(in,"Orientation","Position");
RIOTe.regPrimitive(out,"LightColor","Color");
```

Finally, the developer codes the loop of conditions under which the input primitives are triggered according to readings from the accelerometer and implements how to consume the output primitives by issuing commands to cause the LED array to light up in different colors.

```
### RapEmbedded: Tiles Square code
if(Accelerometer.movement_detected())
  RIOTe.trigger("Orientation",
  Accelerometer.3D_position());

### RapEmbedded: LED Array code
RIOTe.when("LightColor", "green",
LED.set_color("green"));
RIOTe.when("LightColor", "red",
LED.set_color("red"));
```

4.2. Application development and deployment

After the firmware is developed and deployed, each hardware device is autonomous and ready to establish a connection with *RapCloud* to send and receive primitives via the *RapMobile* app, which acts as a gateway. Primitives are now available while coding the application logic using the *RapCloud* online IDE. Back to our example, the developer first registers the application name and the *Virtual Devices* required in the web-based configuration utility. Then she proceeds with coding the application logic. When the first *Orientation* primitive is received, the application logic starts keeping track of the shower time. Water and energy consumption are inferred based on the orientation of the handle. When the consumption of either energy or water exceeds the first configured threshold, the *Light Color* output primitive is triggered to switch the LED array to yellow. If the second threshold is also reached, another *Light Color* primitive is triggered to change the LED array color to red. When the child finishes showering, data about time, energy, and water consumption is uploaded to an online spreadsheet or charting service.

The application code is written directly in the browser using the *RapCloud* online IDE (Fig. 3). When the code is executed, the application is immediately available to end users.

4.3. Application appropriation

Using the *RapMobile* app, the end user performs the *application appropriation* process as described in detail in Section 3.3. She can then start the application directly from the *RapMobile* app.

5. Initial evaluation

Two preliminary evaluations of the system were performed: a pilot test with five computer science university students and a workshop with 14 high school students, aged 15 and 16. The users were asked to develop and deploy an IoT application in less than 60 min. We decided to focus the assessment on the support provided by RapIoT during the implementation of an IoT application, starting from a provided idea. For this reason, and to avoid a complicated application logic, the example application does not include any particular collaborative aspect. All the participants had some experience in programming, although none of them declared to be an expert. Their knowledge of IoT was rather generic and limited, if any; they fit into our definition of non-expert developers in IoT provided in Section 1. Data was collected in the form of observations of the process, answers to a five-point Likert scale questionnaire, and, for the pilot test only, analysis of the source code produced.

The pilot test users were divided into two groups, referenced as A and B. Both groups were able to rapidly get started with application coding and prototyping with no help from workshop supervisors. A scenario describing the IoT application they were asked to develop was provided: *Have you ever been to a party where your shoes have been separated and you can find only one when leaving? To solve this problem, you can implement an application to find a shoe when you have located the other. Double tapping on one shoe will turn on the LEDs on both shoes and vibrate the other shoe. Tilting any shoe will turn off the LEDs on both shoes.*

The code produced by group A of the pilot test can be seen in Listing 1. The groups utilized two separate IoT devices and employed two input primitives (Double Tap, Tilt) and four output primitives (Haptic Long, Haptic Burst, Led On, Led Off). Using the online IDE, both the groups implemented correctly the functionalities described in the scenario provided. The participants were also able to deploy the IoT devices and connect them to *RapCloud* using the *RapMobile* app and the web-based configuration utility. The official documentation available on the web-based configuration utility provided guidance for application deployment and development.

```
var tilesLib = require('/tiles-lib/api');
var client = new tilesLib.TilesClient('Petter', 'Petter_test', 'cloud.rapiot.com', 1883).connect();
var reader = new tilesLib.EventReader();

client.on('receive', function (tileId, event) {
  /* AUTO GENERATED CODE START (do not remove) */
  var shoe_right = reader.getTile('shoe_right', client);
  var shoe_left = reader.getTile('shoe_left', client);
  /* AUTO GENERATED CODE END (do not remove) */
  var tileEvent = reader.readEvent(event, client);

  if (tileEvent.pName == "double tap") {
    shoe_right.trigger("led", "on", "green");
    shoe_left.trigger("led", "on", "green");
    if(tileEvent.name == shoe_left.name) {
      shoe_right.trigger("haptic", "burst");
    } else {
      shoe_left.trigger("haptic", "burst");
    }
  }
  if(tileEvent.pName == "tilt") {
    shoe_right.trigger("led", "off");
    shoe_left.trigger("led", "off");
} });
```

Listing 1: Source code produced by group A during the pilot test.

The two groups adopted different programming strategies to code the behavior. Group A produced a shorter but less robust program than group B, which included additional controls that might have been helpful when extending the application. During the workshop, we tested the same scenario used in the pilot test, providing the same tools and documentation. The participants were younger and less experienced compared to the pilot test. They were divided into four groups; each group had at its disposal a laptop, an Android smartphone running the *RapMobile* app, and several IoT devices. The students were ultimately able to prototype the IoT scenario with little help from workshop supervisors, although they had some difficulties understanding the JavaScript syntax. This issue is not indicative of a systemic problem: RapIoT can retain the same architecture and paradigm while being updated to support more user-friendly high-level programming languages. The adoption of different abstractions such as visual programming languages is a possible extension.

In Fig. 7, we report the results from six of the questionnaire statements: Q5 - the steps of the prototyping process were easy to follow; Q6 - I faced few challenges with the process description; Q8 - during the prototyping process, it was always clear to me what I was supposed to do; Q9 - following the steps of the prototyping process was fun; Q14 - using the *RapCloud* web IDE was easy; Q15 - I faced few challenges using the *RapCloud* web IDE.

Most of the questionnaire answers reported are in the positive end of the Likert scale. The participants of the workshop found the prototyping process slightly more difficult to follow than the pilot test users did. Based on the feedback received and the observed

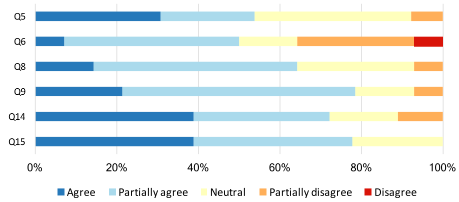


Fig. 7. Questionnaire results.

behaviors, the problem seemed due to the fact that the students, being familiar with the Python programming language, tried to use its syntax when coding in JavaScript.

During the pilot test, we observed the participants struggling with the *RapMobile* application; however, no issues were recorded during the workshop, where a redesigned and updated version of the application was used.

6. Implementation

Here we describe in more in detail the building blocks and technicalities of the RapIoT framework. RapIoT comprises three different stacks of software modules implementing the functionalities provided by *RapCloud*, *RapMobile*, and *RapEmbedded* (Fig. 2).

This design choice spares the implementation of event routing, since each IoT device can be unequivocally controlled by an application running in the cloud no matter where the application or the hardware is deployed. This architecture enables the reuse of deployed devices for different applications without changing the firmware. The development of RapIoT applications is supported by the Cloud9 platform, which allows non-expert developers to create, run, and debug code in a browser. Each user-made application is stored in a user-assigned workspace. The Cloud9 IDE has been extended with a configuration GUI that allows developers to create workspaces, assign them to users, and generate template code directly in a user's workspace. Both the IDE and the configuration GUI (Fig. 3) interact with lower *RapCloud* tiers via a REST interface defined by *RapCloud* APIs, allowing for easy IDE replacement or for usage of multiple IDEs. When writing the logic of a RapIoT application, non-expert developers need only to handle instances of input primitives received from the IoT devices and send instances of output primitives to those devices without the need to know how the modules implement the actual recognition and actuation of primitives. The primitives available to the user within the Cloud9 IDE can be created via the APIs provided by the *RapEmbedded* library. The library implements both registration of primitives and handling of primitives at run-time. The library is written in C++ and built on top of the Arduino IDE. The library is compatible with most Arduino boards that provide BLE connectivity (see Section 3.4). Once primitives are implemented through the *RapEmbedded* library, they can be exchanged back and forth between applications and IoT devices. Instances of output primitives are generated by an application and propagated to a specific IoT device. Otherwise, instances of input primitives are generated by a device and propagated to an application. RapIoT makes use of MQTT as a transport protocol to exchange primitives between *RapCloud* and *RapMobile* stacks. Primitives are coded in JSON-formatted messages that contain a unique identifier for the IoT device, followed by the identifier of the primitive and two optional parameters. The current implementation of the unique identifier allows the IoT device to use a BLE advertised parameter. This guarantees consistency between different mobile OSs, compared to using the Bluetooth MAC address as an identifier. Problems arise in particular with iOS, which scrambles the Bluetooth MAC address, preventing a robust mapping between IoT devices and application logic. An alternative, more reliable solution is to use a UUID,[12] advertised by the IoT device via Bluetooth, before connecting to the mobile app gateway. After an application written using the Cloud9 IDE is started, it begins exchanging instances of primitives on an event-driven basis. Multiple applications can run at the same time inside the server, although one device can exchange primitives with one application only. The JavaScript application created by the developer generates or consumes primitives thanks to the *RapCloud* library and API. The library parses input primitives and triggers JavaScript events that are handled by the developer's code. When an output primitive is triggered, the library takes care of building a well-formed JSON packet and forwarding it to the user's gateway over MQTT. The library makes use of Ponte,[13] to bridge the REST interfaces exposed towards the Cloud9 IDE and administration GUI. Ponte also bridges *RapCloud* with the MQTT interface exposed towards *RapMobile*. Finally, *RapCloud* employs MongoDB[14] to store associations among users, applications, gateways, virtual devices, and real devices. The *RapCloud* stack runs on top of the Node.js JavaScript runtime environment.[15] *RapMobile* bridges IoT devices with applications; the app has two roles. During application appropriation (Section 3.3), it assigns *Virtual Devices* to physical devices in Bluetooth range and locks them within a specific application belonging to a user. This operation is done by the user through the *RapMobile* app (Fig. 4), which is developed using the Ionic Framework.[16] At run-time, *RapMobile* translates MQTT packages containing JSON-formatted primitives into simple comma-separated values that are forwarded to paired devices over a BLE link. Bluetooth connectivity is guaranteed thanks to the BLE library provided by the Cordova BLE Central plugin.[17] *RapMobile* also takes care of error handling when devices move out of Bluetooth range, disconnect from the network, or turn off because they run out of battery power. Concurrent access to a single IoT device, resulting in an access conflict, is prevented by the BLE handshaking mechanism: When a device is connected to a gateway, it stops advertising and accepting further connections. On the gateway, a single IoT application at a time can be started. Furthermore, hardware modules can be discovered, attached to, or removed from the platform while applications are running. Special system-wide events inform connected applications of the availability of new devices in real time.

7. Discussion

In this section, we discuss strengths and limitations of RapIoT in relation to the design goals described in Section 3.1. We then elaborate on how collaboration is supported and why the approach of the toolkit is interesting for smart cities applications.

7.1. Meeting the design goals

Using qualitative assessments and the initial evaluation of the software toolkit presented in Section 5, we now connect the requirements listed in Section 3 with the components of RapIoT. Primitives provide high-level abstraction to encapsulate input and output data packets. Development of plug-and-play software and

12 https://tools.ietf.org/html/rfc4122.
13 http://eclipse.org/ponte/.
14 https://www.mongodb.com/.
15 https://nodejs.org.
16 https://ionicframework.com/.
17 https://github.com/don/cordova-plugin-ble-central.

hardware prototyping platforms based on such high-level object abstractions could mitigate the challenges related to heterogeneity and complexity of Internet of Things network nodes as well as the diversity of modes of communication [38]. In line with the plug-and-play philosophy, RapIoT hides hardware and network complexities (A3, A4), allowing non-expert developers to concentrate the technical effort into the cloud-supported programming phase (B2). On the other hand, more expert developers can extend the primitives supported by the IoT devices through the *RapEmbedded* Arduino library (A1). Primitives permit developers to decouple the application logic, which resides in the cloud, from the rest of the infrastructure tasked with generating, consuming and routing the application-independent primitives across the embedded, gateway, and cloud layers (A2). Thanks to the neutral and multi-purpose nature of the primitives, the architecture is not tied to any specific application domain (B4). Bluetooth-equipped networked sensor nodes can achieve good interoperability with consumer devices, have lower power consumption than WiFi, and have a lower cost (B3) [39]. Bluetooth is also by far the most widespread technology supported by existing consumer devices (B1) [39]. BLE connectivity and the *RapMobile* implementation allow for several IoT devices to be connected to the application layer at the same time (A6). Constraints on the supported IoT hardware are dictated only by Arduino compatibility, which is currently provided by many hardware manufacturers and devices, while new ones are constantly added (A5). Being as both RapIoT and Arduino are open source, community support is a viable medium to share and reuse knowledge (B5). Our approach to IoT system development embeds mechanisms that facilitate the authoring of collaborative applications. Primitives are flexible constructs that allow developers to break down interaction routines and data flows into simpler blocks that can be combined when writing the application logic. The RapIoT toolkit presents four fundamental features that help in the development of collaborative applications:

- *Support for multiple devices* — RapIoT supports applications that make use of several IoT devices connected to the same gateway (C1). This allows multiple users to interact with various devices placed in the same environment, which are then ruled by a centralized application logic running on the *RapCloud* server.
- *HCI primitives for physical interaction* — Some of the primitives rely on composite human actions and events (C3), which involve more than one physical device. It is possible to design and implement applications that support time coordination, sequential actions, awareness, proximity, and other forms of cooperative practices that characterize coordinated ecologies of devices (C4). Under this perspective, the application matches with the definition of a CSCW product: a technology-driven application supporting coordination of collaborative activities [40] and with the notion of *physical collaborative interface* [15].
- *Distributed gateways and devices* — Applications developed with RapIoT can use several gateways physically located in different places, each of which can control a group of devices. This opens these devices up to more flexible scenarios of use: (i) groups of users can move from site to site where different groups of IoT devices are located and perform collaborative tasks that involve IoT devices at the site, e.g., a collaborative treasure hunt game and (ii) users can carry one or more IoT devices connected to their smartphones and perform some tasks or collect data in the environment, remotely cooperating with other users who are following the same workflow but at a different site.
- *Integration with online services* — Connection with third-party collaborative services and online data sets is supported through specific primitives (C2). Asynchronous collaboration is facilitated, allowing users to reflect and cooperate on shared resources connected to the IoT application logic.

To make collaboration happen, users should be engaged in a joint effort, with the ability and flexibility to align their goals and resources with others in real time. All parties should be brought into alignment around what is needed [41]. The technical infrastructure of RapIoT supports this collaborative effort, allowing the creation of collaborative IoT applications addressing a shared goal or challenging and engaging users at different levels.

7.2. RapIoT in a smart city context

Starting from ideas generated by a set of possible end users, the RapIoT architecture has been used to design the technical infrastructure of IoT applications aimed at solving real problems affecting modern smart cities. The simplicity of the approach used, resulting from the architectural choices at the base of RapIoT, can allow developers to prototype and program an initial demo of the wished behavior in a few hours. Future scenarios can involve physical data visualization devices in the city [42] and development of customized primitives tailored to handle smart city sensor data. RapIoT is not limited to any particular domain but is a promising approach able to address the problem of technical stagnation in smart city applications [20].

7.3. Limitations

The RapIoT architecture does not comprehend any coded application logic embedded into IoT devices. Since the primitives have to follow a complete round trip from the embedded layer to the application layer, network latency can be a significant factor affecting performance and application responsiveness. Network quality and availability is crucial for the entire period when the application is in use. This limitation can be particularly amplified when the application layer deals with batches of primitives in rapid sequence. In these cases, most of the execution time is spent waiting for the network, which can hinder the user experience. Another possible limitation is connected to the concept of primitives: for some applications, the behavior to encapsulate in a primitive can be too complex to be exposed with a simple interface like the one provided by input/output primitives. This restriction could be partially mitigated by splitting the logic into two or more primitives, with the drawback of delegating more work to the network.

8. Conclusions

In this paper, we presented the RapIoT toolkit for rapid prototyping of IoT applications. The development of a RapIoT application has been presented by describing the prototyping process of a solution addressing real challenges of smart cities. RapIoT leverages the concept of data primitives as communication blocks and interfaces between generic IoT devices and the application layer. Further, we have highlighted how RapIoT primitives can support the development of collaborative applications via multiple embedded devices, physical interfaces, and distributed gateways. RapIoT takes advantage of and builds on the most recent technological evolutions in the field, such as the Arduino platform, cloud computing, BLE radios, and mobile applications, reducing complexity and entry barriers for non-experts. Compared to the state of the art, with RapIoT we are lowering the adoption threshold by shielding developers from some of the complexity connected to prototyping IoT applications. This process is also facilitated by the holistic nature of the architecture encompassing all three layers of a typical IoT system. In this paper, we have focused on providing an overview of RapIoT and have illustrated through a preliminary evaluation how the development process is supported and facilitated by the toolkit. We have conducted workshops with different users to test the system: Non-expert developers were asked to code and prototype a simple IoT scenario previously presented to them. The groups were able to program the desired behavior and test the final IoT application by physically interacting with the IoT devices employed. As part of our future work, we aim to conduct more systematic studies with users with different levels of programming skill to evaluate what scaffolding mechanisms can be embedded in the toolkit so that it can be used directly by end users with no programming knowledge.

Acknowledgments

We would like to thank all the students that participated in the development of RapIoT, contributing with their work to the publication of this article.

References

[1] O. Eris, J. Drury, D. Ercolini, A collaboration-focused taxonomy of the internet of things, in: 2015 IEEE 2nd World Forum on Internet of Things, WF-IoT, 2015, pp. 29–34.
[2] K. Schmidt, L. Bannon, Constructing CSCW: The first quarter century, Comput. Support. Coop. Work (CSCW) 22 (4–6) (2013) 345–372.
[3] A.J. Jara, M.A. Zamora, A.F.G. Skarmeta, An internet of things–based personal device for diabetes therapy management in ambient assisted living (AAL), Pers. Ubiquitous Comput. 15 (4) (2011) 431–440.
[4] M. Brereton, A. Soro, K. Vaisutis, P. Roe, The messaging kettle: Prototyping connection over a distance between adult children and older parents, in: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, CHI'15, ACM, Seoul, Republic of Korea, 2015, pp. 713–716.
[5] L. Yang, S.H. Yang, L. Plotnick, How the internet of things technology enhances emergency response operations, Technol. Forecast. Soc. Change 80 (9) (2013) 1854–1867.
[6] N. Taylor, U. Hurley, P. Connolly, Making community: The wider role of makerspaces in public life, in: Proceedings of the 2016 CHI Conference on Hum. Factors in Computing Systems, CHI'16, ACM, Santa Clara, California, USA, 2016, pp. 1415–1425.
[7] K. Ashton, That "internet of things" thing, RFiD J. 22 (7) (2009) 97–114.
[8] T.L. Koreshoff, T. Robertson, T.W. Leong, Internet of things: A review of literature and products, in: Proceedings of the 25th Australian Computer-Human Interaction Conference: Augmentation, Application, Innovation, Collaboration, OzCHI'13, ACM, Adelaide, Australia, 2013, pp. 335–344.
[9] M. Blackstock, R. Lea, IoT mashups with the WoTKit, in: 2012 3rd International Conference on the Internet of Things, IOT, 2012, pp. 159–166.
[10] D. De Roeck, K. Slegers, J. Criel, M. Godon, L. Claeys, K. Kilpi, A. Jacobs, I would DiYSE for it!: A manifesto for do-it-yourself internet-of-things creation, in: Proceedings of the 7th Nordic Conference on Human-Computer Interaction: Making Sense Through Design, NordiCHI'12, ACM, Copenhagen, Denmark, 2012, pp. 170–179.
[11] S. Mora, J. Asheim, A. Kjøllesdal, M. Divitini, Tiles cards: A card-based design game for smart objects ecosystems, in: Proceedings of the First International Workshop on Smart Ecosystems cReation By Visual dEsign Co-Located with the International Working Conference on Advanced Visual Interfaces, Vol. 1602, (AVI 2016), CEUR-WS, Bari, Italy, 2016, pp. 19–24.
[12] A. Crabtree, T. Rodden, S. Benford, Moving with the times: IT research and the boundaries of CSCW, Comput. Support. Coop. Work (CSCW) 14 (3) (2005) 217–251.
[13] B.A. Farshchian, M. Divitini, Collaboration support for mobile users in ubiquitous environments, Handb. Ambient Intell. Smart Environ. (2010) 173–199.
[14] S. Bødker, Third-wave HCI, 10 years later—participation and sharing, Interactions 22 (5) (2015) 24–31.
[15] S. Greenberg, Toolkits and interface creativity, Multimedia Tools Appl. 32 (2) (2007) 139–159.
[16] A. Botta, W. De Donato, V. Persico, A. Pescapé, On the integration of cloud computing and internet of things, in: 2014 International Conference on Future Internet of Things and Cloud, (FiCloud), IEEE, 2014, pp. 23–30.
[17] J. Grudin, S. Poltrock, Computer supported cooperative work: the encyclopedia of human-computer interaction, Encyclopedia Hum.-Comput. Interact. (2012).
[18] T. Zachariah, N. Klugman, B. Campbell, J. Adkins, N. Jackson, P. Dutta, The internet of things has a gateway problem, in: Proceedings of the 16th International Workshop on Mobile Computing Systems and Applications, ACM, 2015, pp. 27–32.
[19] I. Wagner, M. Basile, L. Ehrenstrasser, V. Maquil, J.-J. Terrin, M. Wagner, Supporting community engagement in the city: Urban planning in the MR-tent, in: Proceedings of the Fourth International Conference on Communities and Technologies, ACM, 2009, pp. 185–194.
[20] F. Gianni, M. Divitini, Technology-enhanced smart city learning: A systematic mapping of the literature, IxD&A 27 (2016) 28–43.
[21] R.G. Hollands, Will the real smart city please stand up? City 12 (3) (2008) 303–320.
[22] S. Mora, F. Gianni, M. Divitini, RapIoT toolkit: Rapid prototyping of collaborative internet of things applications, in: 2016 International Conference on Collaboration Technologies and Systems, (CTS), IEEE, 2016, pp. 438–445.
[23] A. Millner, E. Baafi, Modkit: Blending and extending approachable platforms for creating computer programs and interactive objects, in: Proceedings of the 10th International Conference on Interaction Design and Children, IDC'11, ACM, Ann Arbor, Michigan, 2011, pp. 250–253.
[24] D. Mellis, M. Banzi, D. Cuartielles, T. Igoe, Arduino: An open electronic prototyping platform, in: Proceedings of CHI Extended Abstracts, ACM, 2007, pp. 1–11.
[25] J. Maloney, M. Resnick, N. Rusk, B. Silverman, E. Eastmond, The scratch programming language and environment, ACM Trans. Comput. Educ. (TOCE) 10 (4) (2010) 15–16.
[26] J. Sadler, K. Durfee, L. Shluzas, P. Blikstein, Bloctopus: A novice modular sensor system for playful prototyping, in: TEI'15: Proceedings of the Ninth International Conference on Tangible, Embedded, and Embodied Interaction, ACM, 2015, pp. 347–354.
[27] S. Sentance, J. Waite, S. Hodges, E. MacLeod, L. Yeomans, Creating cool stuff: Pupils' experience of the BBC micro: Bit, in: Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education, ACM, 2017, pp. 531–536.
[28] W. McGrath, M. Etemadi, S. Roy, B. Hartmann, Fabryq: Using phones as gateways to prototype internet of things applications using web scripting, in: Proceedings of the 7th ACM SIGCHI Symposium on Engineering Interactive Computing Systems, ACM, 2015, pp. 164–173.
[29] Q. Zhu, R. Wang, Q. Chen, Y. Liu, W. Qin, IOT gateway: Bridging wireless sensor networks into internet of things, in: 2010 IEEE/IFIP 8th International Conference on Embedded and Ubiquitous Computing, (EUC), 2010, pp. 347–352.
[30] S. Nastic, S. Sehic, M. Vögler, H.-L. Truong, S. Dustdar, PatRICIA–A novel programming model for IoT applications on cloud platforms, in: 2013 IEEE 6th International Conference on Service-Oriented Computing and Applications, IEEE, 2013, pp. 53–60.
[31] F. Khodadadi, R.N. Calheiros, R. Buyya, A data-centric framework for development and deployment of internet of things applications in clouds, in: 2015 IEEE Tenth International Conference on Intelligent Sensors, Sensor Networks and Information Processing, (ISSNIP), 2015, pp. 1–6.
[32] M. Kovatsch, S. Mayer, B. Ostermaier, Moving application logic from the firmware to the cloud: towards the thin server architecture for the internet of things, in: 2012 Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, (IMIS), IEEE, 2012, pp. 751–756.
[33] A. Pintus, D. Carboni, A. Piras, The anatomy of a large scale social web for internet enabled objects, in: Proceedings of the Second International Workshop on Web of Things, ACM, 2011, p. 6.
[34] K. Schmidt, Divided by a common acronym: On the fragmentation of CSCW, in: ECSCW 2009, Springer, 2009, pp. 223–242.
[35] J. Gubbi, R. Buyya, S. Marusic, M. Palaniswami, Internet of things (IoT): A vision, architectural elements, and future directions, Future Gener. Comput. Syst. 29 (7) (2013) 1645–1660.
[36] O. Shaer, E. Hornecker, Tangible user interfaces: Past, present, and future directions, Found. Trends Hum.-Comput. Interact. 3 (1–2) (2009) 1–137.
[37] S. Mora, F. Gianni, M. Divitini, Tiles: A card-based ideation toolkit for the internet of things, in: Proceedings of the 2017 ACM Conference Companion Publication on Designing Interactive Systems, DIS'17 Companion, ACM, Edinburgh, Scotland, 2017, pp. 587–598.
[38] I.P. Cvijikj, F. Michahelles, The toolkit approach for end-user participation in the internet of things, Architecting Internet Things (2011) 65–96.
[39] P.P. Pereira, J. Eliasson, R. Kyusakov, J. Delsing, A. Raayatinezhad, M. Johansson, Enabling cloud connectivity for mobile internet of things applications, in: 2013 IEEE 7th International Symposium on Service Oriented System Engineering, (SOSE), IEEE, 2013, pp. 518–526.

[40] P.H. Carstensen, K. Schmidt, Computer supported cooperative work: New challenges to systems design, in: K. Itoh (Ed.), Handbook of Human Factors, Citeseer, 1999, pp. 619–636.
[41] R. Ashkenas, There's a difference between cooperation and collaboration, Harv. Bus. Rev. (2015).
[42] S. Houben, C. Golsteijn, S. Gallacher, R. Johnson, S. Bakker, N. Marquardt, L. Capra, Y. Rogers, Physikit: Data engagement through physical ambient visualizations in the home, in: Proceedings of the 2016 CHI Conference on Hum. Factors in Computing Systems, ACM, 2016, pp. 1608–1619.

Francesco Gianni is a Ph.D. student in Information Systems and Software Engineering group within the Department of Computer Science at the Norwegian University of Science and Technology. His research focuses on CSCW, IoT and ubiquitous computing applied to smart cities, to promote sustainability and lifelong learning in the citizens. His recent research has focused on user centered ideation, design and prototyping of IoT applications.

Simone Mora holds a postdoctoral fellowship at the department of Computer Science, NTNU. He graduated in 2015 with a thesis on leveraging sensor-based interaction for supporting reflection at work, with crisis training as case study. His research investigates the development of methods and tools for ideation and rapid prototyping of Internet of Things ecologies. He applies object augmentation as a design strategy and embraces participatory and user-centric design. He co-authored more than 20 peer-reviewed papers in the fields of computer-supported reflective learning, collaboration technologies, tangible interfaces and HCI. In his recent past Dr. Mora has been a visiting scholar at City London University and MIT.

Monica Divitini is professor of Cooperation Technologies at NTNU. She holds a Master in Information Science from Milano University, Italy and a Ph.D. in Computer Science from Aalborg University, Denmark. She has more than 20 years active experience in research with focus on technology enhanced learning, cooperation technology, mobile and ubiquitous computing, user-centered design approaches. She investigates innovative learning methods enhanced by technology, with focus on training outside the classroom and in the workplace through e.g. collaborative game-based and reflective learning.

Paper 7

Rapid Prototyping Internet of Things Applications for Augmented Objects: The Tiles Toolkit Approach

Francesco Gianni, Simone Mora, and Monica Divitini

In: Ambient Intelligence (AmI)