Léo Leplat

Generic Platforms for Indicator Visualization in City Planning

Master's thesis in Simulation and Visualization
Supervisor: Ricardo da Silva Torres
June 2022

Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of ICT and Natural Sciences

Kunnskap for en bedre verden

Léo Leplat

Generic Platforms for Indicator Visualization in City Planning

Master's thesis in Simulation and Visualization
Supervisor: Ricardo da Silva Torres
June 2022

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of ICT and Natural Sciences

NTNU
Norwegian University of
Science and Technology

Generic Platforms for Indicator Visualization in City Planning

Léo Leplat

Abstract

Increasingly more urban data have become available in the last decade, thanks to the wide use of sensing technologies and several governmental initiatives. These data can be useful for different applications, such as urban planning based on key performance indicators. Their proper use, however, requires effective visualization tools to support correct and insightful understanding and analysis. Much research has been carried out on modeling and integrating geographical data, and different tools have been developed for their visualization. However, those tools are usually application-dependent and cannot be easily tailored to other problems. Moreover, some technical knowledge is often required to use them properly. This thesis introduces the design, implementation, and validation of two flexible indicator visualization and assessment platforms in city planning. The architecture of applications supporting the visualization and the assessment of indicators based on spatial data is presented, and then implemented for use by both urban planners and citizens. Several case studies involving the assessment of bus service availability, walkability, and city mobility in the city of Ålesund, Norway, are discussed. Then, two evaluations with groups of users are performed to validate the use of the applications. Obtained results demonstrate that the developed tools are flexible and can be used by urban planners and citizens. The developed tools are available at `https://github.com/Rylern/master-thesis` – As of June 2022.

Acknowledgments

I would like to thank my supervisor Ricardo da Silva Torres, who has provided consistent support and assistance throughout my work. His supervision encouraged me to complete this master thesis.

I would also like to thank Claudia Viviana Lopez Alfaro for her help and advice regarding the development of the applications, the organisation of the mobile application workshop, and the writing of the thesis.

The workshop of the desktop application would not have been possible without the help of Andreas Amundsen and Dina Margrethe Aspen. I am thankful for their assistance in organizing the arrangements of the evaluation.

Finally, I would like to thank the 18 participants of the evaluations. Their active participation and feedback made the evaluation work possible.

This work was developed as part of the NORDARK project, funded by Nord-Forsk, and the Smart Plan [grant number #310056] and Twin Fjord [grant number #320627] projects funded by the Research Council of Norway.

Preface

This document contains a master thesis of the Master of Science degree at the Simulation and Visualization program at the Norwegian University of Science and Technology (NTNU). This project was done during the spring of 2022 and explores the design, implementation, and validation of generic platforms for multidimensional variable (indicator) visualization to support city planning activities.

Contents

Figures

Tables

Acronyms

BIM Building information modeling. 7

CAD Computer-aided design. 7

CPU Central Processing Unit. 59

DBMS Database Management System. 11

GIS Geographic Information System. 5, 7, 18, 20, 32, 38, 47, 48, 60–62

JSON JavaScript Object Notation. 30, 38

OGD Open Government Data. 1

REST API REpresentational State Transfer Application Programming Interface. 11, 18, 29, 38

SDK Software Development Kit. 18, 38

TIN Triangulated irregular network. 41

VR Virtual Reality. 7

Chapter 1

Introduction

The general background of the conducted research is described in Section 1.1. Then, the objectives and contributions are presented in Section 1.2. Finally, the outline of the thesis is provided in Section 1.3.

1.1 Background

An increasing number of sensors deployed as well as the wide use of social media platforms have fostered the creation of large datasets. More and more data have also become publicly available during the last years as part of several governmental initiatives, such as the Open Government Data (OGD) movement [1]. For example, the number of datasets on the European Data Portal[1] has increased from 400 000 in 2016 to 800 000 in 2019. These data are often related to education, health, budget classification, political boundaries, transportation, and crime reports [2], and thus can be very useful for different applications. For example, they can help policy-makers during the urban planning processes by highlighting the characteristics of different neighborhoods of a city. Citizens may also use these data in everyday life. For example, available data could be used to find a route between two locations considering different factors, such as weather or traffic conditions.

However, technical expertise is required to understand, process, and visualize raw data. Moreover, such information is often too voluminous to be understood with a single look on it. Therefore, effective visualization tools are needed to support the proper understanding and analysis of such data. Such platforms have to follow specific requirements. First, data may vary depending on the specifications of one location or one problem, so the tool has to be flexible. Second, the visualization of data has to be done in a short and clear way, and has to be suited for the target user. For example, urban planners would want to have different parameters to customize the visualization for their specific use case, while citizens would prefer a simpler and easier solution to understand conveyed information. Finally,

[1] `https://data.europa.eu/en` (As of May 2022).

different users may utilize distinct equipment to process and visualize data. For example, urban planners may be more favorable to use powerful computers and large screens that enable the display of large amount of data, while citizens may prefer the use of mobile applications with simpler and customized interfaces. Examples of such applications can be found in [3–5].

A good example of such visualization tools are map-based interfaces. In these applications, data are displayed on top of 2D or 3D maps. Examples can be found in navigation applications (e.g., Google Maps[2], Waze[3]), food delivery applications (Uber[4]), weather applications (Weather forecast[5]), and more. Map-based interfaces are used when there is a need to provide a visualization of spatial data, i.e., data which directly or indirectly references a specific geographical area.

This research work is concerned with the investigation of map-based visualization tools in the context of urban planning applications that target different user groups. Applications implemented included the assessment of walkability and accessibility of different neighborhoods in Ålesund municipality, Norway. Walkability refers to how friendly a region is to walking activities [6], while accessibility refers to the availability of specific places or services for an individual at a given location [7].

Several indicators can be used to assess both walkability and accessibility. Indicators can be understood as the variables or elements in the urban area that influence the levels of walkability and/or accesibility in a given time and space. These variables or indicators can function alone or as a combination. For example, the emphasis can be laid on safety by considering the car speed limit or the level of artificial illumination. Walkability and accessibility aspects have become crucial in urban planning given their association with health benefits [8] and quality of life. Furthermore, fostering walking habits and giving more accessibility may reduce traffic congestion, air pollution, gas emissions, and the dependence on fossil fuels for transportation [8]. Therefore, policy-makers often want to improve both walkability and accessibility of regions during planning processes.

1.2 Objectives and Contributions

Several tools focusing on the visualization of spatial indicators have already been developed [3, 9, 10], including some specifically built to address analysis in urban planning [6]. However, such tools are application-dependent and thus require changes in the source code to address other problems. Therefore, the main goal of this thesis was to design, implement, and validate flexible indicator visualization platforms addressing the following research questions:

[2] `https://www.google.com/maps` (As of May 2022).
[3] `https://www.waze.com/live-map/` (As of May 2022).
[4] `https://www.uber.com/no/en/` (As of May 2022).
[5] `https://play.google.com/store/apps/details?id=com.chanel.weather.forecast.accu&hl=en&gl=US` (As of May 2022).

- **RQ1:** How to design a generic platform that supports both the visualization and the assessment of indicators based on spatial data?
  A review of existing spatial data visualization methods is made by looking at their functional description, architecture, and technologies used. Then, new generic platform architectures that handle the visualization and the assessment of any spatial indicators are proposed.
- **RQ2:** How to implement applications based on the generic platforms that can be easily used by different target users?
  Target applications may be useful for different types of users, for example urban planners or citizens. They have different goals and equipment, so an investigation is carried on which platform and device would be more suited for each target user. Then, an appropriate implementation of the platforms is proposed.
- **RQ3:** Would the use of created tools be effective to support the analysis of indicators for different scenarios?
  The proposed platforms are instantiated in different tools, which focus on several case studies related to urban planning. In particular, walkability, accessibility, and availability of bus services were assessed, in the city of Ålesund, Norway.
- **RQ4:** Would the developed tools be useful for different user groups?
  Two evaluations were conducted after the implementation of the applications on case studies related to walkability and accessibility analysis. The goal was to determine the relevance and functionality for the target users. The usability of the application is validated through this evaluation.

Three indicator visualization tools were designed and implemented in this research:

- GENOR, a generic visualization platform that supports decision making in urban areas. The platform handles the definition, storage, processing, and visualization of indicators associated with an area in 2D and 3D map-based views. The goal of this platform is to help policy-makers making the right decisions by providing a clear and direct visualization of indicators. This tool can be used for any kind of indicator-based analysis. Some examples are walkability, accessibility, and public transport assessment.
- A web-based visualization tool focused on the municipality of Ålesund, Norway. The application lets the user customize the definition of walkability or accessibility and then provides a visualization of the indicator(s) on top of roads. It also has a navigation feature, which can be used to find a route between two locations, taking into consideration restrictions such as slope. This tool uses some of the principles and implementation aspects of GENOR, but is more focused on walkability and accessibility. It is meant to help policy-makers during the planning process.
- A mobile visualization application that is similar to the web-based tool, but intended for citizens as users.

One validation was performed on the web-based visualization tool and another for the mobile visualization application. In both workshops, the participants had the possibility to use the applications and give feedback through forms.

1.3 Outline of Thesis

The remaining of this document is organized as follows:

- Chapter 2 presents a review of existing tools and methods used in urban visualization.
- Chapter 3[6] describes the design and implementation of a generic visualization platform, named GENOR, that can be seamlessly tailored for different indicator-based analyses.
- Chapter 4 presents a flexible indicator visualisation platform developed for city planners and citizens.
- Chapter 5 describes two evaluations conducted on the developed tools.
- Chapter 6 presents a discussion on the work that has been done during this thesis as well as the answers to the research questions.

[6] Part of the content of this chapter was presented in a paper published in the proceedings of the 36th ECMS International Conference on Modelling and Simulation (2022) [11].

Chapter 2

Related work

Much research has been carried out on analyzing and visualizing spatial urban data. This chapter provides an overview of recent data modeling, integration approaches, and relevant urban visualization methods.

2.1 Geographic Information Systems

A milestone in mapping, territorial studies, and urban planning was the development of Geographic Information System (GIS) and their commercialization for microcomputers in the 1980s. A Geographic Information System is an information system designed to collect, store, process, analyze, manage, and present all types of spatial and geographic data. Spatial data can be classified into two types: raster and vector data. Raster data consists of rows and columns of cells, each cell containing a value. Vector data consists of points, line, polygons, or a combination of these elements, each type containing a vector value. GIS-related applications are tools that allow users to create interactive queries, analyze spatial information, and modify and edit data through maps. They can be used for land use planning, infrastructure and network management, transport and logistics, insurance, telecommunications, engineering, planning, education and research. Due to the improvements of hardware and advances in geographic algorithms, GIS are now widely used in urban planning not only as a database but also as a toolbox for analysis and visualization of results [12].

In the late 1990s, a new pivotal concept appears, Public Participation GIS (PPGIS). PPGIS describes a more people-centered GIS that allows citizens to take part in public policy and planning processes, unlike traditional expert-driven GIS [13]. By combining GIS with other fields, such as Web 2.0 technologies, some applications were developed and proved to be a valuable approach for engaging the public [14, 15]. Such tools can promote communication among users and decision makers by being easy to set up and use [16]. Recent initiatives can be found in [17], where community mapping tools were used in the city of Tijuana, Mexico to identify environmental risks, and in [18], where children used a GIS software and completed questionnaires that helped the author to determine a relationship between housing characteristics and children's independent mobility.

2.2 Integration of geographical data and data modeling

Data modeling and integration refer to the combination of data coming from different sources into a single unified view. One stream of literature has focused on it, especially on working with inconsistent and heterogeneous data. For example, Fortini & Davis [9] presented methods that enable integration and visualization of urban data coming from multiple heterogeneous sources. Their platform is divided into integration, data storage, and service providing parts. The integration module is responsible for combining information from multiple heterogeneous data sources by performing data extraction, preprocessing, and standardization on each type of data, resulting in files with only one format suited for storage.

Psyllidis et al. [3] presented a web-based platform that supports the analysis, integration, and visualization of large-scale and heterogeneous urban data. Because datasets are usually specific to one sector or domain, establishing relationships between these data can turn to be a difficult task. Their platform aims to solve this problem by using a semantic enrichment and integration component in the form of a web-based interface. It allows the user to define the relations between urban systems, data sources, and the city technology enablers.

Chen et al. [19] described an urban data visualization tool that focuses on the cross-domain correlation from multiple data sources by providing selection, filtering, and aggregating features. This is done using a visual query model for cross-domain correlation and a visual analytic framework for urban data visualization, correlation, querying, and reasoning. Similar strategies for handling different types of indicators were adopted in this thesis, but with less liberty given to the user in order to create simpler applications.

2.3 Initiatives on urban data visualization

Several initiatives have been proposed to support the proper visualization of urban data, including for mobility and transportation analysis. For example, Feng et al. [20] used density maps to assess the accessibility measurement for public service facilities, while Bello et al. [21] designed interactive 3D visualizations to analyse urban noise pollution. For a review in the area, the reader may refer to [22]. A generic discussion about indicator projects is made in [23], especially on how they influence the governance of cities. The authors found that such initiatives improve and make more transparent the decision making process, but are also not neutral because of their contingencies and relationalities.

Eberhardt & Silveira [2] presented a list of visualization techniques applied to Open Government Data. They found that map-based presentation is the most used visualization method, and the main use case of developed visualization tools refer to studies related to transportation issues. A similar vision regarding the importance of map-based visualization strategies was reported in [24]. Their work described an infrastructure to collect data and a map-based visualization method to display and understand tourist flows. Their tool helped communities foster awareness about sustainability issues, but it was only designed for one purpose and study area.

In another venue, Sauda et al. [25] provided an interactive urban visualization tool that displays multiple dimensions of urban data, allowing the user to have a understanding of the urban environment. Today, most visualization methods for geographical data are in 2D. However, visualizations in 3D can help to increase the understanding of spatial relations. Johansson et al. [26] provided methods that highlight social values through 3D visualization, while J. Döllner et al. [27] described a tool able to combine different application domains (GIS, Computer-aided design (CAD), and Building information modeling (BIM)) to obtain a 3D visualization of a city. New technologies, such as Virtual Reality (VR), can also be effective when creating urban visualization. For example, Perhac et al. [10] described another urban data visualization method that uses VR with a game engine (Unity) to display data in a more immersive way. For a review of the capacity of VR in urban planning, the reader may refer to [28].

Some generic GIS applications that can be used for specific problems have been also developed. The ArcGIS Urban software,[1] made by ESRI,[2] has been used to create several initiatives related to smart city planning. For example, the city of Uppsala (Sweden) used it to create a visualization of the city, helping planners to create a new district that improves residents' quality of life, does not subtract from biodiversity or degrades the environment, and cuts carbon emissions.[3]

[1] `https://www.esri.com/en-us/arcgis/products/arcgis-urban/overview` (As of May 2022).
[2] `https://www.esri.com/en-us/home` (As of May 2022).
[3] `https://www.esri.com/about/newsroom/blog/uppsala-sustainable-development/` (As of May 2022).

Chapter 3

GENOR: A Generic Platform for Indicator Assessment

Platforms to visualize spatial indicators have been previously designed [3, 6, 9, 10]. But those mostly focus on one specific problem or domain and thus can not be easily tailored for different applications or different users. For example, Longva [6] designed a platform for the assessment of walkability in the city of Ålesund, Norway. Using a client / server architecture, this tool allows the user to select and combine walkability indicators and then presents walkability results on top of a 2D map.

This chapter describes a generic visualization platform that can be seamlessly tailored for different indicator-based analyses. The goal of this platform was to provide clear indicator visualizations that support planners and policy-makers in decision-making in urban planning. This platform, called GENOR, handles the definition, storage, processing, and visualization of indicators associated with areas using 2D and 3D map-based views.

Two case studies were considered to validate the proposed platform, both related to problems in urban mobility analysis in Ålesund:

- First, the platform was used to evaluate the walkability index in the city of Ålesund. Walkability ca be defined as a city's attractiveness or its opportunity for walking [29].
- Second, the differences in the bus services across different districts of the city were assessed.

These two cases provide examples on how this platform can support urban planning by assessing the use of public spaces and the effectiveness of public transportation services.

3.1 Architectural description

This section introduces the fundamental structure of GENOR.

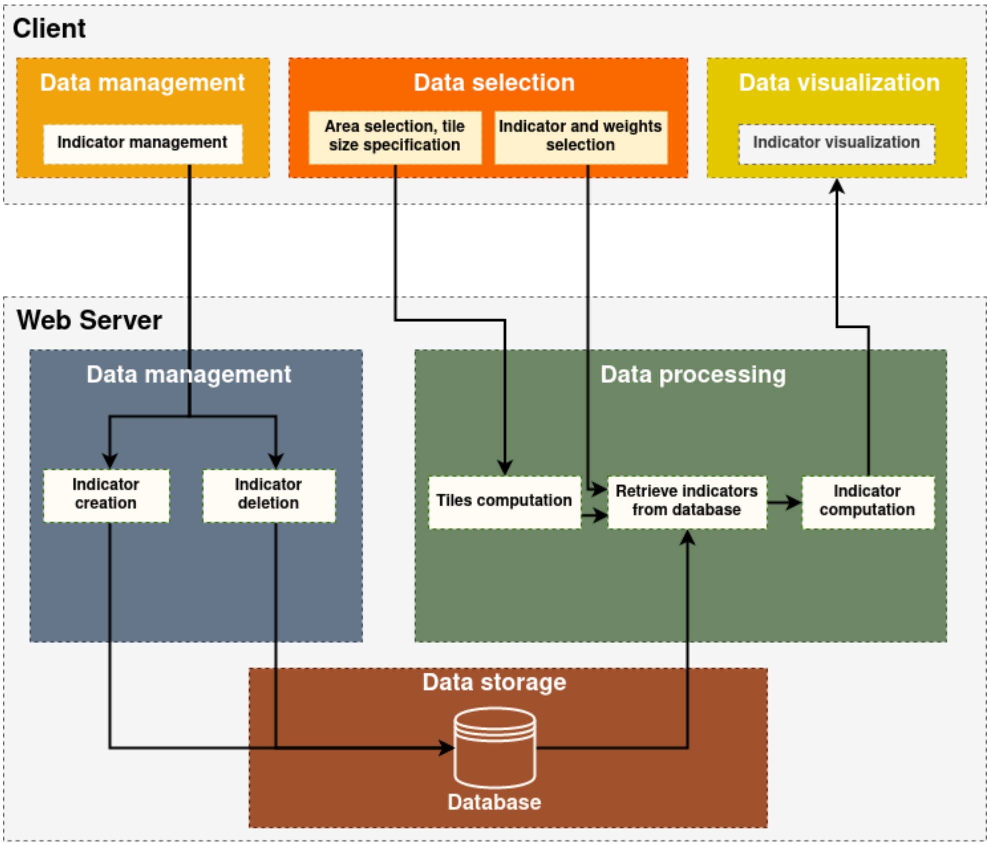


Figure 3.1: Architectural view of the GENOR platform.

GENOR is divided into two entities: a client, to handle the user interaction and visualization, and a web server, to handle the data storage and processing. Figure 3.1 presents the architecture of GENOR.

3.1.1 Client side

The client side handles all the interactions with the users, allowing them to define indicators and visualize computed indicator scores for a particular area. This side has three components:

- The data management part, which handles the definition of the indicators to be visualized. Indicators can also be deleted if they are no longer needed. Once the user defines or deletes indicators, the client application will send these data to the server which will process the information.
- The visualization of indicators, in which users must first choose an area where the indicators will be displayed, and then the indicators to be visualized. If multiple indicators are selected, a feature to assign weights to each indicator is included. The client application will send these data to the server, and the server will respond with the computed indicators.

- The data visualization part, which happens when the server responds to the data selection request by sending values of indicators for specific locations. The client application will create one or more visualization methods to show these data efficiently based on this information.

3.1.2 Server side

The server is responsible for storing and processing data. The user has no direct interaction with it, but the server is essential for the client application to work properly.

- The data management part handles the definition and deletion of the indicators to be visualized. This information is sent to the data storage module.
- The data processing part handles the computation of indicators. Based on an area and indicators selected by the user, the server communicates with the data storage module to compute the specified indicators for that particular region.
    - First, the region of interest is divided into multiple tiles whose size is given by the user. The goal is to compute one indicator value for each tile.
    - Then, the server communicates with the database to retrieve the relevant indicators of the region of interest.
    - Finally, based on the previous data, the server assigns a value of all indicators to each tile. For each tile, all indicators are merged into one global indicator using a weighted linear sum, the weights being defined by the user. The server can then send a list of tiles with one value for each of them to the client application.
- The data storage module contains a Database Management System (DBMS) that is used to store indicators. Even if it is represented within the web server, the database could be physically separated from it.

Usually, indicators are grouped when an analysis is made. For example, a walkability assessment combines multiple indicators, such as the number of pedestrians crossings or the average speed limit in the neighborhood. Therefore, the indicators were grouped by *category*. An indicator is a value assigned to an area, and a category is a group of one or more indicators.

The web server is a REpresentational State Transfer Application Programming Interface (REST API). A REST API is an API that conforms to the constraints of the REST architectural style. REST is a software architectural style commonly used to create interactive applications that use web services. The client uses keywords to make requests: GET to retrieve resources; POST to submit new data to the server; PUT to update existing data; and DELETE to remove data.

Figure 3.2 presents the database diagram used in this work. A table *indicator* keeps track of the indicators. Each table row consists of a unique identifier and the indicator name. A table *category*, similar to the *indicator* table, is used to keep

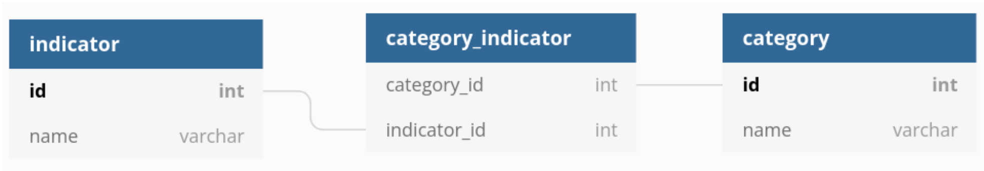


Figure 3.2: Database diagram.

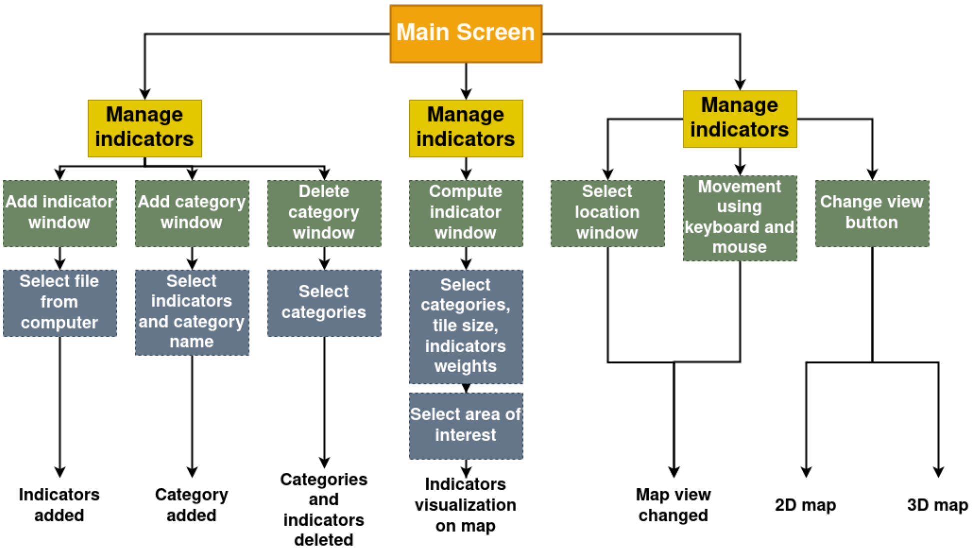


Figure 3.3: Functional flow diagram of the platform.

track of the categories. To determine the indicators a category possesses, a third table *category_indicator* was created. Each row contains the identifier of a category and the identifier of an indicator.

3.2 Platform features

This section presents the features of GENOR. Figure 3.3 presents a functional view of the client interface. Three actions can be performed from the main screen: managing indicators, computing indicators, and changing the map display.

3.2.1 Managing indicators

The first action the tool offers refers to the management of indicators. This means adding or deleting indicators and categories.

Adding indicators

When the user wants to add an indicator, the corresponding window opens (Figure 3.4). This feature allows the user to choose a local file that contains values of

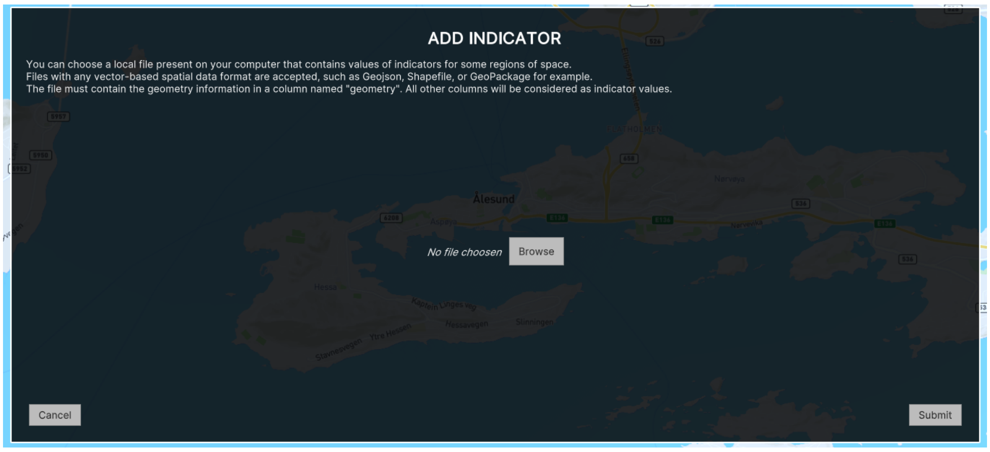


Figure 3.4: Add indicator window.

Table 3.1: Example of an indicator table.

geometry	indicator
POLYGON [(62.473940, 6.178904), (62.473940, 6.190553), 62.472451, 6.178904), (62.472451, 6.190553)]	0.5
POLYGON [(48.098141, -1.378394), (48.098141, -1.341239), (48.080142, -1.378394), (48.080142, -1.341239)]	0.7

indicators for some regions of space. To make the platform as generic as possible, files with any vector-based spatial data format are accepted, such as Geojson,[1] Shapefile,[2] or GeoPackage[3] for example. The file is then sent to the server through a POST request.

The server reads the file and converts it to a generic format. Once performed, the server will request the database to create a table for each indicator present in the file and populate it with the corresponding values. Table 3.1 shows an example with two rows. The table has an indicator column that contains the indicator value, and a geometry column that contains an area corresponding to that indicator value. The coordinates follow the EPSG:4326[4] format.

Adding categories

A category is a group of indicators. Therefore, the "add-category" window (Figure 3.5) shows a list of all available indicators. To have access to this list, the client sends a GET request to the server before actually opening the window. The user can define a category name, choose one or more indicators and send this information to the server through a POST request.

With the information the client application provides, the server will then create a row with the category name to the *category* table. The category identifier,

[1] `https://geojson.org/` (As of May 2022).

[2] `https://www.esri.com/content/dam/esrisites/sitecore-archive/Files/Pdfs/library/whitepapers/pdfs/shapefile.pdf` (As of May 2022).

[3] `https://www.geopackage.org/` (As of May 2022).

[4] `https://epsg.io/4326` (As of May 2022).

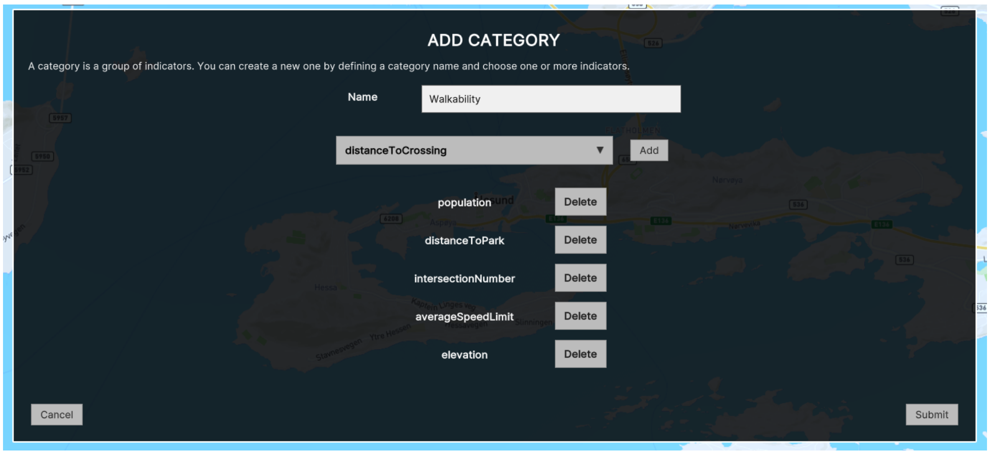


Figure 3.5: Add category window.

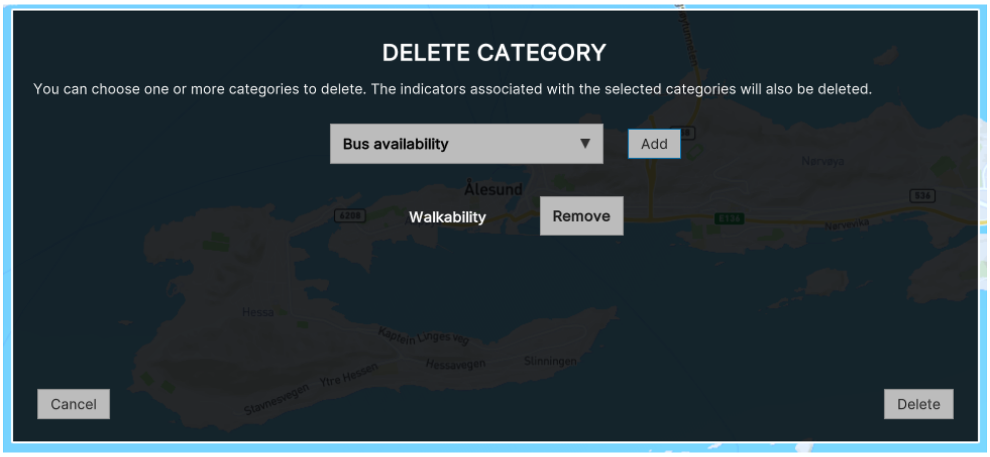


Figure 3.6: Delete category window.

automatically created, is sent to the server. Then, for each indicator the client wants in this category, a row will be created in the *category_indicator* table with the previous category identifier and the identifier corresponding to the indicator.

Deleting categories

The delete category window (Figure 3.6) shows the list of available categories. This list is given by the server after a GET request by the client. Then, the user can choose one or more categories, and the selection will be sent to the server through a DELETE request.

The server will remove each category and its associated indicators from the database. This means that some rows will be removed from the *indicator*, *category*, and *category_indicator* tables, and the tables containing the indicators will be dropped.

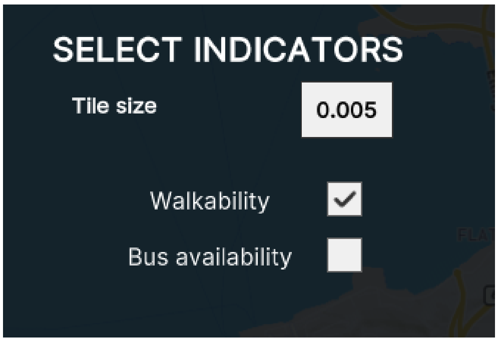


Figure 3.7: Part of page 1 of the “compute-indicator” window.

3.2.2 Computing Indicators

Once some indicators and categories are defined, the user can start computing and visualizing them for the selected area. This area is divided into several tiles, where one indicator value is computed. The client application can then provide a visualization of the results.

Client side

The “compute-indicator” window (Figures 3.7 and 3.8) allows the user to choose which indicators to visualize. On the first page of the window, a list of available categories is displayed (retrieved from the server through a GET request), as well as an input to specify the size of the tiles. On the second page of the window, a list of indicators corresponding to the selected categories is displayed. The user can specify the weights associated with each of the indicators. Then, the user is asked to select an area on the map in which the indicators will be computed. Finally, all of these data (selected categories, tile size, weights of indicators, selected area) are sent to the server through a GET request.

The server computes the indicators (as described in the next part) and sends back tiles with values of indicators that can be visualized in the client application.

The current version of the platform provides two visualization methods:

- A 2D visualization, where each tile is represented by a square whose color changes depending on the indicator’s value. A high value is represented by the green color, while a low value is represented by a red color. This visualization method is commonly used by urban data visualization tools [2].
- A 3D visualization, where some relief and 3D buildings are added to the map. Each tile is represented by a vertical bar, whose height and color depend on the indicator’s value, like the previous visualization. 3D visualizations offer more understanding of spatial relations than 2D visualizations when the user can freely move in the environment [30]. Furthermore, 3D models are considered to be more beneficial for citizens, planners, and politicians [31].

Figure 3.8: Part of page 2 of the “compute-indicator” window.

Server side

When the server is asked to compute the indicators, three steps are conducted: First, the server creates the tiles by dividing the selected area based on the tile size. The centroid (or center) of the tiles is also computed. Second, for each indicator, the server tries to obtain the values of this indicator for that specific area from the database. Each indicator has a corresponding table inside the database, in which the values of this indicator are attached to areas. Then, a spatial join between the centroids of the tiles and the data taken from the database is made. If a centroid of a tile is located inside an area of the indicator table, then the server can assign the corresponding indicator value to this tile.

Figure 3.9 shows an example. An indicator table contains two rows and a table representing the tiles. Before the spatial join, the tiles have no value for the indicator. After the join, the first row gets the value 0.2 because its centroid is located inside the first polygon of the indicator table. The second row still has no value because its centroid is not located inside any polygon.

Finally, one global indicator is computed for each tile by combining all indicators, as shown in Equation 3.1, with ω_i the weights and v_i the indicator values.

$$globalIndicator = \sum_{i \in \{indicators\}} \omega_i \times v_i \tag{3.1}$$

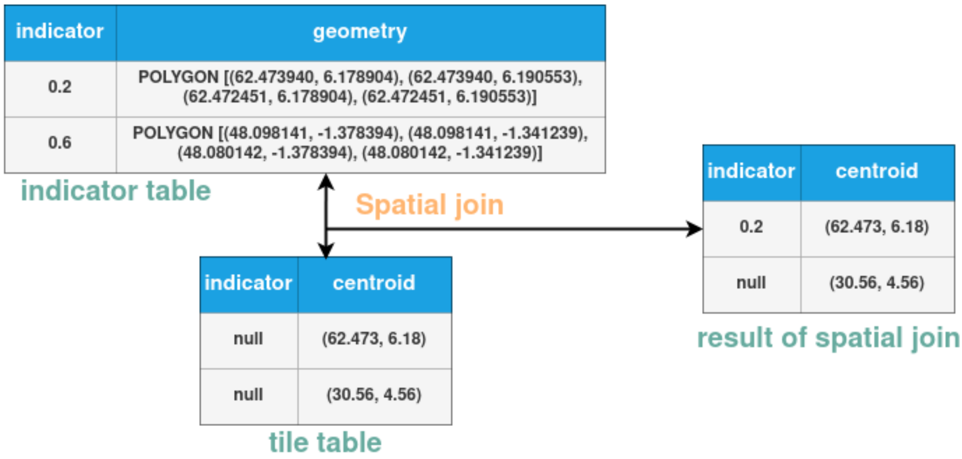


Figure 3.9: Example of a spatial join.

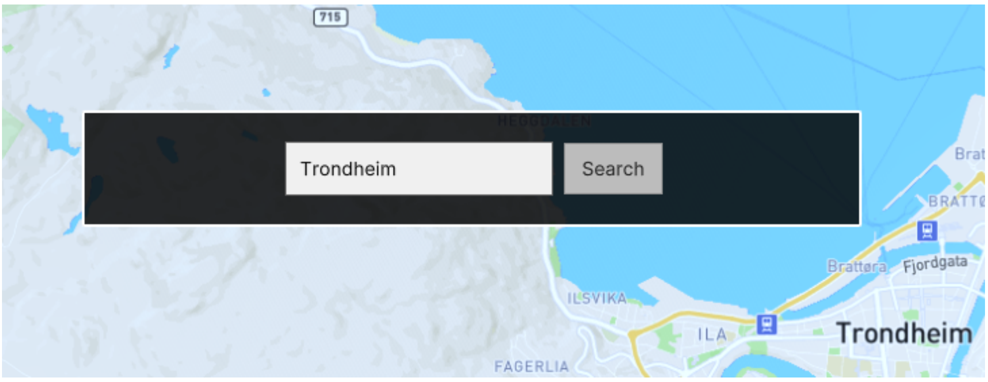


Figure 3.10: Change location window.

3.2.3 Changing the Map Display

Two map views are available: a 2D view and a 3D view. It is possible to pan using the mouse in the 2D map. The 3D view presents a view of projected satellite photos on terrain with elevation and 3D buildings. The mouse and the keyboard can be used to move inside the scene.

Finally, a window (Figure 3.10) is available to support a quick change of the location. The user can input a place, e.g. a city, and the map will be moved to that location.

3.2.4 Implementation aspects

Our implementation considered usage scenarios that rely on 2D and 3D visualization methods on top of a world map. Therefore, the client side was implemented using a Unity[5] application. Unity is a cross-platform game engine that can be used

[5] `https://unity.com/` (As of June 2022).

to create 2D and 3D games and interactive simulations.

The utilization of a map provider is needed to display an interactive map. Many exist, such as ArcGIS,[6] Google Maps,[7] or Geopipe.[8] Mapbox[9] was adopted because it provides a well-documented Software Development Kit (SDK) for Unity.

The server side consists of two components: the web server and the database. The requirements for the server were to create a REST API that can work with GIS data. Creating a REST API can be done easily using different tools, such as JavaScript (Node.js) and PHP. Python-based technologies were adopted for handling GIS data, because of the availability of several GIS packages. Therefore, the web server is written in Python. To create the REST API, the package FastAPI[10] was used, which is a modern and fast web framework for building an API using Python.

Several GIS packages are used, the most important ones being:

- Geopandas,[11] to work with geospatial data.
- Shapely,[12] to manipulate and analyze planar geometric objects.
- Pyproj,[13] to perform cartographic transformations and geodetic computations.

The database had a similar requirement than the web server: it had to support GIS data. Postgresql[14] was adopted, which is a free and open-source relational database management system, along with PostGIS,[15] which is an open source software that supports handling geographic objects in PostgreSQL.

The implementation diagram is shown on Figure 3.11.

3.3 Case studies

This section presents two case studies related to the use of GENOR for the visualization of indicators related to urban mobility, and a comparison of the platform with an existing Digital Twin for Walkability Assessment in City Planning.

3.3.1 On the visualization of walkability indicators

The first case study was the visualization of the walkability index for the city of Ålesund in the context of planning processes.

[6] https://developers.arcgis.com/unity-sdk/ (As of May 2022).

[7] https://developers.google.com/maps/documentation/gaming/overview_musk (As of May 2022).

[8] https://geopi.pe/games (As of May 2022).

[9] https://www.mapbox.com/unity (As of May 2022).

[10] https://fastapi.tiangolo.com/ (As of May 2022).

[11] https://geopandas.org/ (As of May 2022).

[12] https://shapely.readthedocs.io/en/stable/manual.html (As of May 2022).

[13] https://pypi.org/project/pyproj/ (As of May 2022).

[14] https://www.postgresql.org/ (As of May 2022).

[15] https://postgis.net/ (As of May 2022).

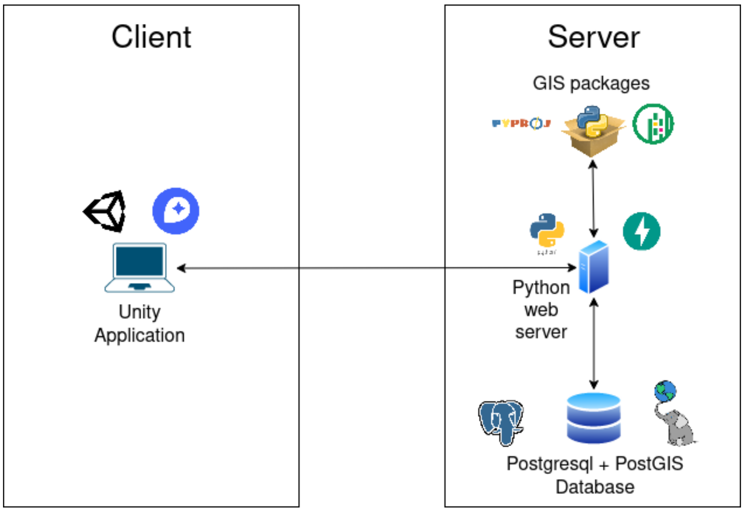


Figure 3.11: Implementation diagram, illustrating the main technologies employed in the implementation of the proposed platform.

Several indicators can be used to compute walkability. Based on the available information, the following indicators were used:

- Population density: defined as the number of people in a grid of 400×400m; higher density means a more walkable area.
- Park areas: the indicator is higher when a park is nearby.
- Street connectivity: defined as the number of street intersections, more street intersections gives a higher score of walkability.
- Elevation: the highest score is at the lowest altitude.
- Speed limit: defined as the road sleep limit, lower speed limit gives a higher score.
- Pedestrian crossings: The indicator is higher when a pedestrian crossing is nearby.

Datasets used

Global datasets have been used to compute the indicators. This means that even though this use case is limited to the city of Ålesund, it can be easily extended to any other place of the world. Table 3.2 presents the datasets used.

Area of interest

The indicators will be computed within an area of interest. For this example, the study area is the city of Ålesund, therefore the area is a bounding box with the

[16] `https://data.humdata.org/dataset/kontur-population-dataset` (As of May 2022).
[17] `https://www.kontur.io/` (As of May 2022).
[18] `https://www.opentopodata.org/` (As of May 2022).
[19] `https://www.openstreetmap.org/` (As of May 2022).

Table 3.2: Datasets used.

Indicator	Dataset	Source	Dataset description
Population density	Kontur population dataset[16]	Kontur[17]: a geospatial data and real-time risk management solutions provider for humanitarian, private, and governmental organizations.	Free, released in 2020, and consists of hexagons with population counts at 400m resolution.
Elevation	Advanced Spaceborne Thermal Emission and Reflection Radiometer (ASTER) dataset through the Open Topo Data API.[18]	Joint effort between the Ministry of Economy, Trade, and Industry (METI) of Japan and the National Aeronautics and Space Administration (NASA) of the US.	1 arc-second resolution, corresponding to a resolution of about 30m at the equator.
Park areas Street connectivity Speed limit Pedestrian crossings	OpenStreetMap[19]	Volunteers	OpenStreetMap is a free, editable map of the whole world that is being built by volunteers.

Figure 3.12: Area of interest.

following coordinates:

- Minimum longitude: 5.938799.
- Minimum latitude: 62.436930.
- Maximum longitude: 6.420250.
- Maximum latitude: 62.536570.

This corresponds to the area in green in Figure 3.12.

Indicator assessment

When creating indicators, the platform expects one or more files with any vector-based spatial data format, such as Geojson, Shapefile, or GeoPackage for example. Therefore, this type of file needs to be created with the previous indicators. Python-based libraries have established as promising technologies to perform such GIS operations.

Population density The population dataset consists of hexagons with population counts at 400m resolution. For each hexagon within the area of interest, one value of all indicators was computed. The Geopandas package is used to read the population dataset by taking only the hexagons located inside the area of interest. The resulting data is then projected onto the EPSG:3857 projection system, the projected coordinate system used for rendering maps in, e.g., Google Maps and OpenStreetMap. Since indicators must take values between 0 and 1, the density indicator is computed as shown in Equation 3.2.

$$densityIndicator = \frac{population}{maxPopulation} \tag{3.2}$$

$maxPopulation$ refers to the maximum value of population inside the area of interest.

Park areas The parks of Ålesund are given by OpenStreetMap through the Osmnx package. It is projected onto the EPSG:3857 projection system to be consistent with the hexagons. First, each hexagon's centroid (or geometric center) is computed. Then, the distance to park parameter is set for each hexagon as the distance between the centroid and the nearest park within walking distance. The walking distance is considered to be 800 meters, the farthest radial distance based on a ten minutes walk.[20] If a centroid has no park within walking distance, the associated parameter equals the walking distance. Finally, since the park areas indicator is higher when a park is nearby, the park area indicator is set as shown in Equation 3.3.

$$parkAreaIndicator = 1 - \frac{distanceToPark}{walkingDistance} \tag{3.3}$$

Street connectivity As for the park areas, the number of street intersections is given by OpenStreetMap through Osmnx. For each hexagon, the number of intersections whose distance to the centroid is inferior to the walking distance is counted. The street connectivity indicator is set as shown in Equation 3.4.

$$streetConnectIndicator = \frac{numberOfIntersect}{maxNumberOfIntersect} \tag{3.4}$$

Elevation The elevation data is coming from Open Topo Data. For each hexagon, the elevation of the centroid is queried. The associated indicator is written in Equation 3.5.

$$elevationIndicator = 1 - \frac{elevation}{maxElevation} \tag{3.5}$$

[20] `https://www.dcla.net/blog/walkability-standards` (As of May 2022).

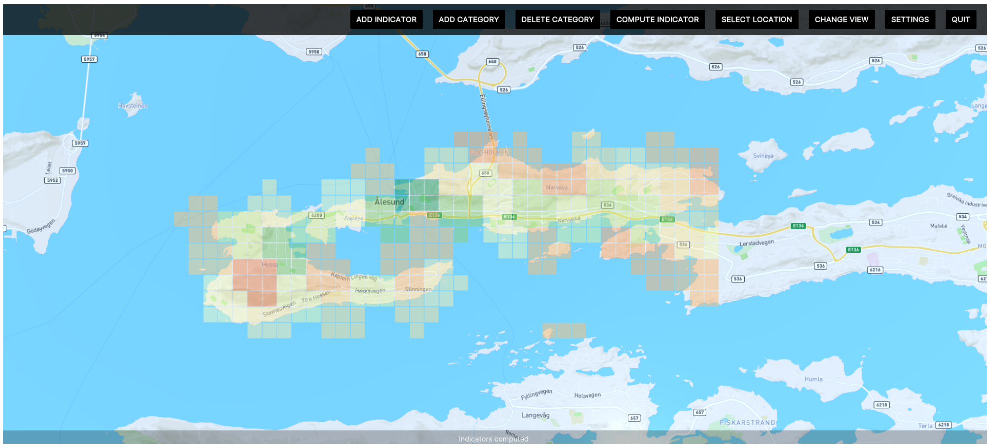


Figure 3.13: 2D visualization of walkability.

Speed limit The speed limit data comes from OpenStreetMap. For each hexagon, an average of the speed limit of each road whose distance to the centroid is inferior to the walking distance is made. The speed limit indicator is written in Equation 3.6.

$$speedLimitIndicator = 1 - \frac{averageSpeedLimit}{maxAverageSpeedLimit} \tag{3.6}$$

Pedestrian crossings The pedestrian crossings indicator computation is similar to the park area indicator computation, the only change is that the distance to the nearest crossing is used instead of the nearest park.

The GeoPandas package was used to export the values of the six indicators to a GeoJSON file. This file format is an open standard format designed for representing simple geographical features.

Visualization of indicators

In the platform, the previously created GeoJSON file can be uploaded via the “add indicator window.” Then, a *walkability* category can be defined in the “add category window.” Finally, the “compute-indicator” window will lead to the visualization of the indicators.

Figures 3.13 and 3.14, respectively, show the 2D and 3D results with all weights set to 1. A green color means a high value, and a red color means a low value. The study case shows that the walkability indicator is the highest in the city center and the lowest in the southwest part of the city.

Indicators can also be visualized individually. For example, Figure 3.15 shows only the 2D result of the population indicator. The red tiles indicate areas with virtually no people living there.

Figure 3.14: 3D visualization of walkability.

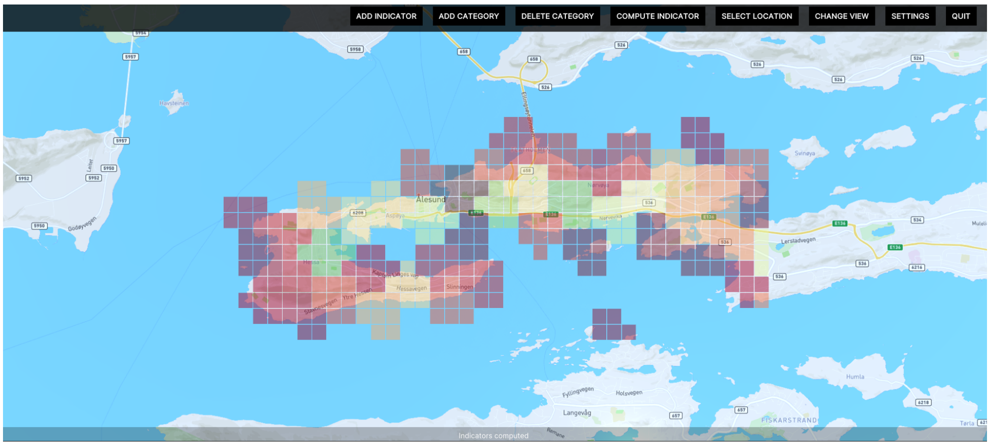


Figure 3.15: 2D visualization of the population indicator.

3.3.2 On the visualization of bus service availability

The second case study is bus service availability in the city of Ålesund. The Public Transport Access Level (PTAL) [32] was computed for different districts of Ålesund and at different intervals of time.

Datasets used

Ålesund was divided into districts based on data coming from Statistics Norway.[21] Figure 3.16 shows the districts of Ålesund.

Data related to buses (bus stops, frequency) was collected from Entur,[22] which operates the national registry for all public transport in Norway.

[21]`https://kart.ssb.no` (As of May 2022).
[22]`https://developer.entur.org/stops-and-timetable-data` (As of May 2022).

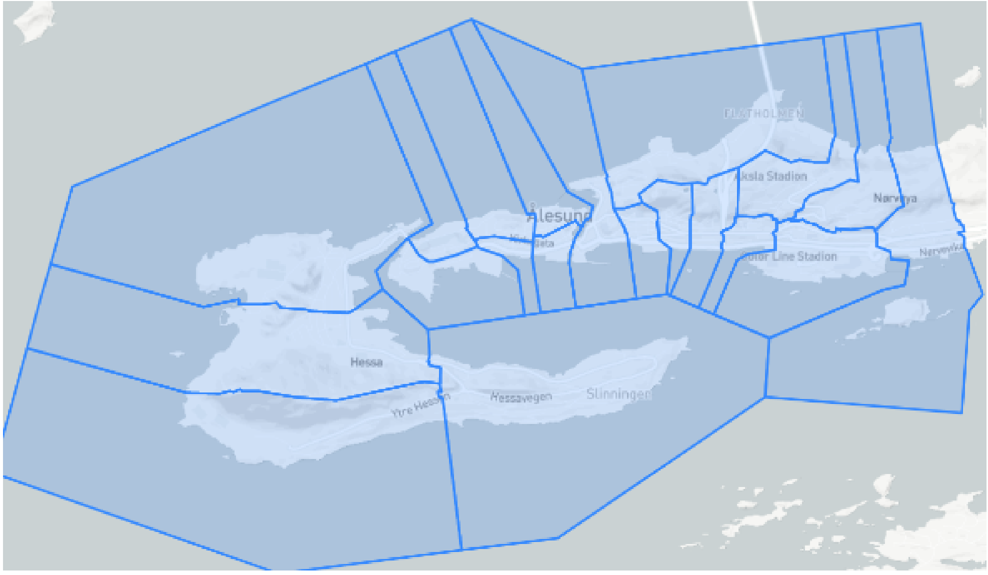


Figure 3.16: Districts of Ålesund.

Indicator creation

To determine the PTAL for one district, the PTAL was computed for all buildings inside that district and then averaged the result. The building data were provided by Mapbox.[23] For each location, the PTAL is computed following an algorithm described in [32]:

- Calculate the walk times from the location to the nearest service access points (bus stops in this example). Only bus stops within 640 meters are considered, because it is assumed that people will walk up to eight minutes to a bus service.[24]
- For each bus stop, calculate the scheduled waiting time (SWT), which is half the time interval between arrivals of buses at this stop. The scheduled waiting time indicator reflects the frequency of buses arriving at a specific stop.
- For each bus stop, calculate the average waiting time (AWT), which is the SWT added to a reliability factor. The reliability factor reflects the fact that actual wait times can be longer due to buses arriving late. It is set to two minutes in this use case.
- For each bus stop, calculate the total access time (TAT), which is equal to the walk time added to the AWT.
- For each bus stop, calculate the equivalent doorstep frequency (EDF), which is a measure of what the service frequency would be like if the service was available without any walking time. It is equal to $EDF = 0.5 \times (\frac{60}{TAT})$.

[23] `https://www.mapbox.com` (As of May 2022).

[24] `https://content.tfl.gov.uk/connectivity-assessment-guide.pdf` (As of May 2022).

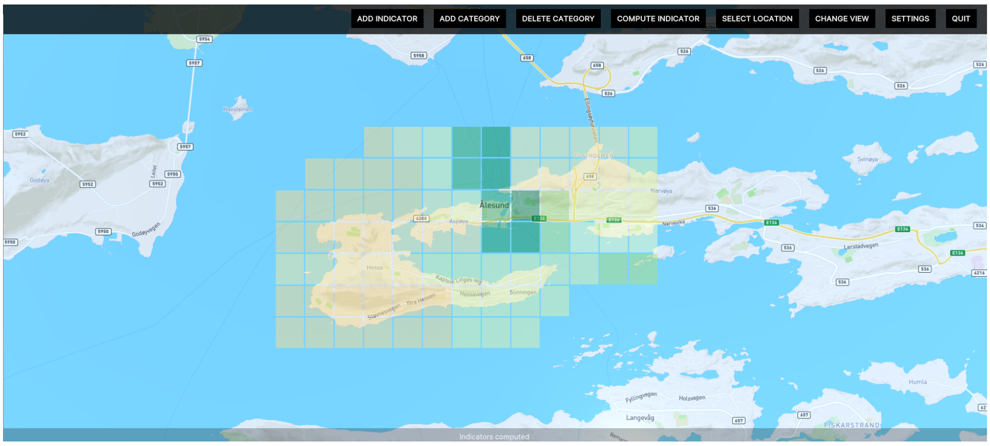


Figure 3.17: 2D visualization of the bus service availability.

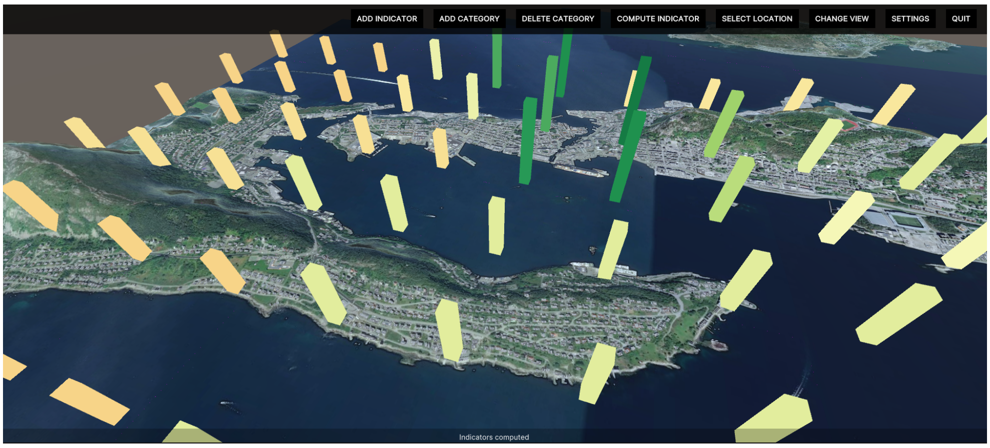


Figure 3.18: 3D visualization of the bus service availability.

- Finally, calculate the Access Index (AI) by summing all the EDF. This gives a value from 0 (worst PTAL) to 40 (best PTAL).

Visualization of indicators

Figures 3.17 and 3.18 show, respectively, the 2D and 3D result with all weights set to 1. A green color means a high value and a red color means a low value, so like the walkability indicator, the bus service availability indicator is the highest in the city center, and the lowest in the south west of the city.

3.3.3 Comparison with the Digital Twin for Walkability Assessment in City Planning

The walkability assessment presented in Case Study 1 was already implemented in [6]. In this section, the two platforms are compared (Table 3.3).

Table 3.3: Comparison of GENOR with the proposal in [6].

	Proposal in [6]	GENOR
Architecture	Client - Server	Client - Server
Client application	Unity	Unity
Server application	Python web server (REST API)	Python web server (REST API)
Data storage system	File system	Database
Computation of indicators	Pre-computed and stored	User has to provide a file
Indicators	Walkability	Any, defined by the user
Indicator selection	Weight selection	Indicators and weights selection
Tile size	Fixed	Can be changed
Area of interest	Ålesund	Any area
Views	2D	2D and 3D

In term of architecture and implementation, both platforms are similar. They use a client server model with Unity and Python. Only the storage system is different (file system and database).

In term of features, the platforms differ. The platform of [6] was made to only visualize walkability in Ålesund, so it is less generic than GENOR. Recall that in the case of GENOR, the user has to generate the indicators, while the platform proposed in [6] assumes their pre-computation. Moreover, our platform presents more ways to customize the visualization with the tile size specification, and the area and indicator selection.

Chapter 4

Generic Platform for indicator visualization for mobile and desktop environments

The second goal of this thesis was to design and implement a flexible indicator visualisation platform for city planners and citizens. These groups have different objectives and needs when using maps and geographic information. Policy-makers would use the application during the planning process. Citizens require information that is effective and easy to use in the everyday life. They may also have different devices at their disposal. Based on the different requirements, two applications have been developed: a desktop application, aimed for urban planners, and a mobile application, aimed for citizens.

Desktop and mobile applications have three main features:

- The user can select indicators that are be displayed on top of roads.
- The application provides a steepness visualization, which indicates which roads can be used by a disabled or elderly person.
- A navigation feature can be used to find a route between two locations. Only the route with the best indicator score is shown to the user.

4.1 Architectural description

This section introduces the fundamental structure of the applications. Since the two applications have similar features and only differ by their implementation, the applications will be referred as the client in this section. Figure 4.1 presents the architecture of the project. As for GENOR, there are two entities: a client to handle user interaction and visualization, and a web server for data processing and storage.

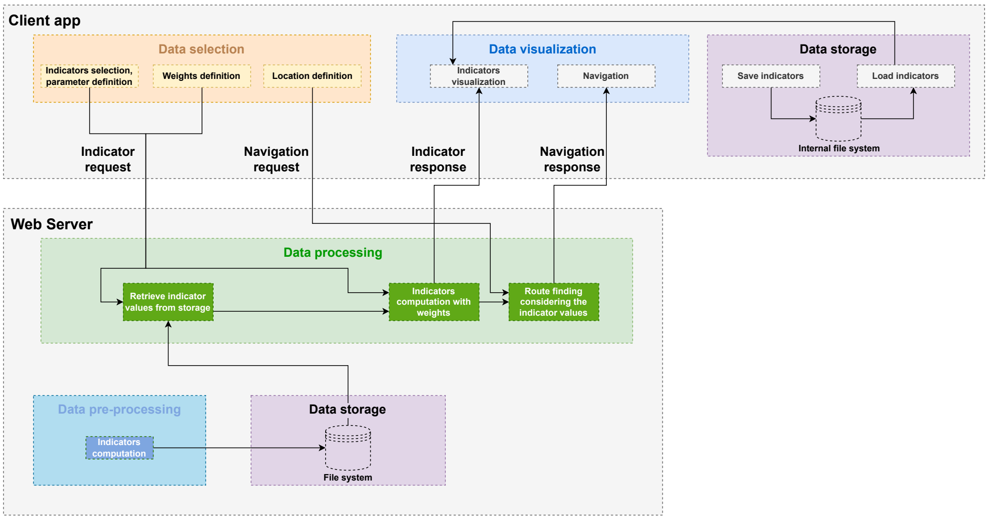


Figure 4.1: Architectural view.

4.1.1 Client side

The client handles all the interaction: the user can select some indicators, and the applications will provide related visualization. A navigation feature allows the user to select two locations, and the application will display a route between those two locations. Three main parts can be described:

- The data selection part lets the user choose what kind of data to visualize. First, indicators have to be selected. Then, some indicators may have a parameter that will change the indicator computation. For example, in the context of the steepness indicator related to walkability, different users may define distinct values of maximum uphill steepness they consider walkable. This type of parameter has to be defined by the user. Finally, in case multiple indicators are selected, the user can define a weight applied to each indicator. For example, a user may define steepness as more important as the proximity of parks. Once all this information is given, the client will send these data to the server. To use the navigation feature, the user can also define two locations that will be sent to the server for processing.
- The data visualization part occurs when the server responds to a request made in the data selection part. As described before, there are two requests: one for visualizing indicators, and the other for finding a route between two locations. The client will create suitable visualizations for these two features.
- The data storage part allows the client to save indicators on its device, so that it can load them afterwards. An internal file system is used to save the data. This part was developed in order to let the user save some indicator configurations considered important. When indicators are loaded, no request to the server is made, so the visualization happens faster.

4.1.2 Server side

As described in the previous part, the client makes two requests to the server: one to compute indicator values, and the other to find a route between two locations. Like GENOR, the web server is a REST API that can accept GET or POST requests for example. To process these requests, three parts can be described:

- First, raw indicator values must be computed. This is performed in the pre-processing step. An area of interest is selected and raw indicator values are computed inside that area. For example, let us consider an indicator related to population. The raw indicator values will be the population density for different locations.
- The raw indicator values are stored in the data storage part. I chose to use a file system, which means the raw indicator values are stored into a file. Other information, such as descriptions for each indicator, is stored into another file.
- The data processing part handles the two requests by using the data stored.
    - When the user makes the indicator request, two steps are conducted. First, the indicators selected by the user are retrieved from the file system. Then, the indicator values are computed using the raw indicator values and parameters defined by the user, and combined using the weights also defined by the user. The result is one overall indicator with values for the area considered, which is sent back to the client. For example, with an indicator related to population, we already saw that the raw indicator values are the population. The corresponding indicator values will be numbers between 0 and 1, 0 for locations without population, and 1 for densely populated locations.
    - When the user makes the navigation request, the steps described above also occur. Then, when the server has computed an overall indicator, a route is determined using the indicator values. For example, if two routes are possible, only the one with the highest indicator value will be considered. When a route is found, it is sent to the client.

4.2 Applications features

This section describes the features of the desktop and the mobile applications. They have two main features: managing indicators and finding a route between two locations.

4.2.1 Indicator management

The first action the tool offers refers to the management of indicators. This means selecting indicators, editing parameters and weights, as well as saving and loading indicators.

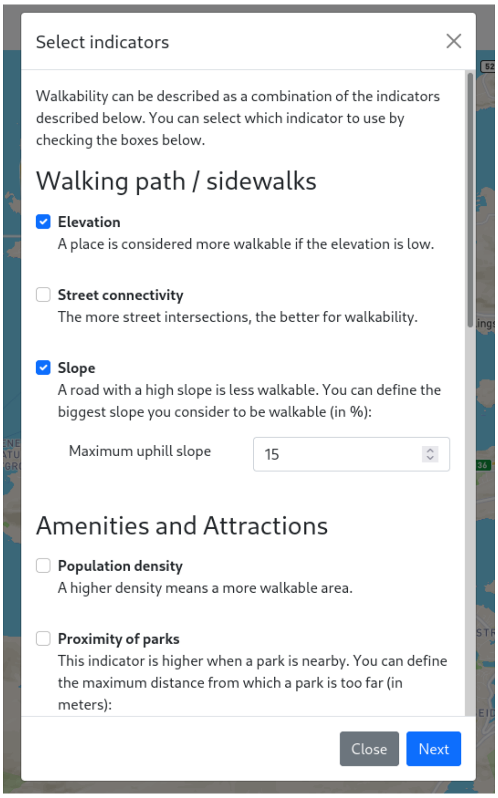


Figure 4.2: Select indicator window (desktop application).

Selecting indicators and parameters

When the user wants to select indicators and parameters, a request is made to the server to fetch information about all indicators, such as the indicator name, a description, and a group name. The server stores this information in a JavaScript Object Notation (JSON) file.

The mobile application has an "advanced mode" feature that only displays some of the indicators. This has been added to avoid overwhelming users with lots of indicators. In case this feature is disabled, the server will only send a part of all the indicators.

After the request has been processed by the server, a window (for the desktop application, Figure 4.2) and a screen (for the mobile application, Figure 4.3) opens, that lets the user define which indicator to use by checking or unchecking boxes. Some indicators, for example slope in Figure 4.2, have an additional parameter that will have an impact during the computation of the indicator values.

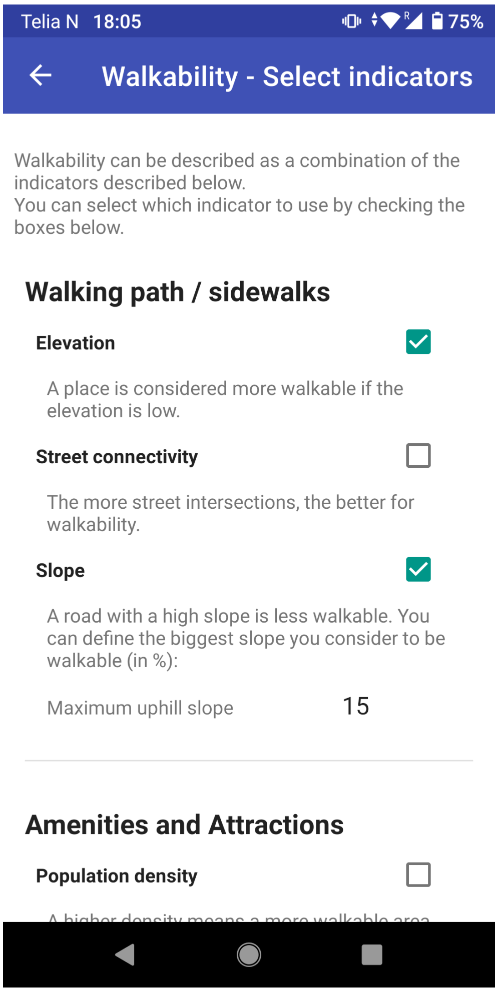


Figure 4.3: Select indicator screen (mobile application).

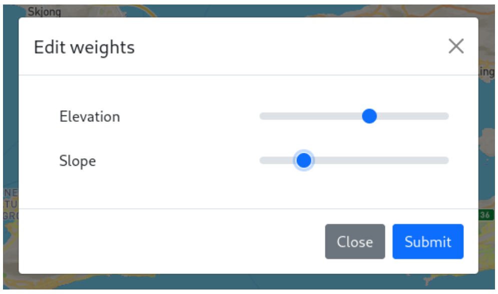


Figure 4.4: Edit weights window (desktop application).

Editing weights

Once indicators are selected, a window (for the desktop application, Figure 4.4) and a screen (for the mobile application, Figure 4.5) let the user define the weights applied to each indicator by moving a slider. The client applications can now send a request to the server for computing indicator values.

Computing indicators

On the server side, computing indicators is done in two steps: fetching indicator raw values from the data storage and computing the indicator values based on the user input.

Indicator values are information related to specific location on Earth. To work with such spatial data, GIS must be used. In this project, vector data are used to represent indicator values (which are described in the case study section).

Once the raw indicator values are loaded, the parameters and weights given by the user are used to compute one overall indicator. The same equation as the one used by GENOR (Equation 3.1, with ω_i the weights and v_i the indicator values), is used to combine the indicators. When the computation is done, the server sends a response to the client with the overall indicator values.

Visualizing indicators

To provide a visualization of the overall indicator values, both applications display a 2D map with a heatmap drawn on top of it (Figure 4.6 for the desktop application, Figure 4.7 for the mobile application). The color scale goes from orange (low value) to yellow (medium value) and to green (high value).

Saving and loading indicators

The client applications have saving and loading features. They allow the user to save indicators on its device so that they can be loaded afterwards.

Due to differences in the technology used, the mobile and desktop applications do not exactly have the same implementations:

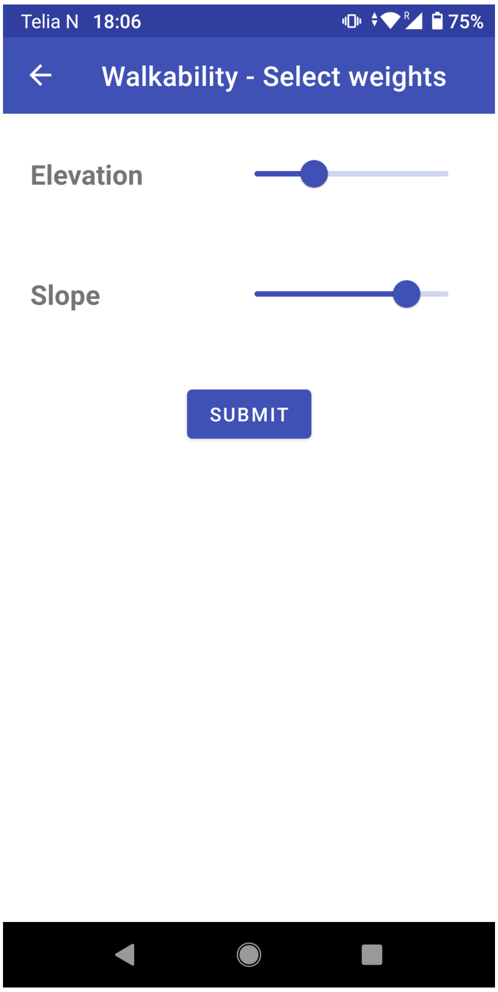


Figure 4.5: Edit weights screen (mobile application).

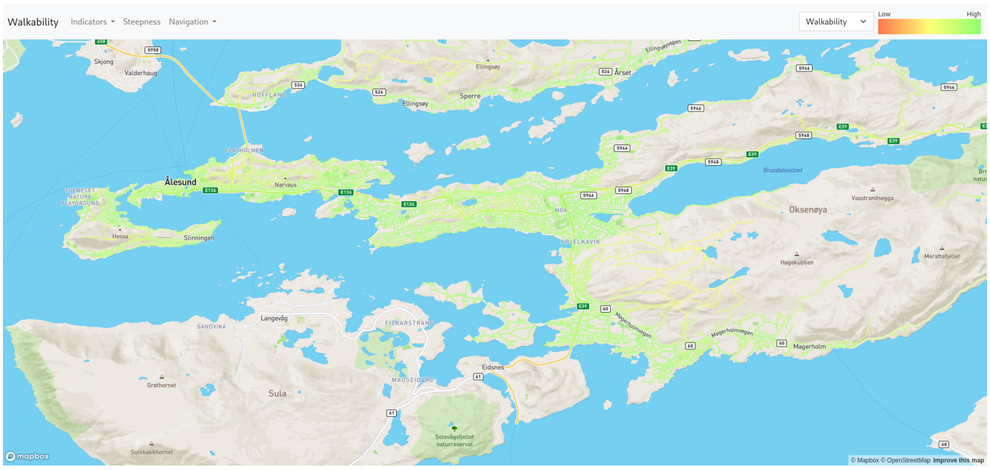


Figure 4.6: Indicator visualization (desktop application).

Figure 4.7: Indicator visualization (mobile application).

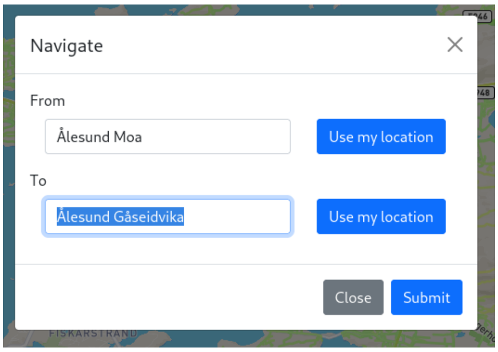


Figure 4.8: Navigation window (desktop application).

- The mobile application saves the indicator values. That can represent large files (more than a few megabytes), but it means that when the user loads the indicators, they are directly displayed without having to make a request to the web server.
- The desktop application saves which indicators are selected, the parameters, and the weights. The indicator values are not stored, because the technology used (which is detailed in the implementation section) does not allow files larger than a few megabytes to be stored. It means that when the user loads the indicators, a request to the server has to be made with the stored parameters and weights.

4.2.2 Navigation

The navigation window (for the desktop application, Figure 4.8) and the navigation screen (for the mobile application, Figure 4.9) let the user define two places. The device's location can be used as one of the two places. The user inputs place name, while the server wants to receive geographic coordinates. Therefore, a conversion has to occur before sending a request to the web server.

Then, the server performs two steps:

- First, an overall indicator is computed as described in the computing indicators section.
- Then, a shortest path algorithm takes as an input the two locations defined by the user and the overall indicator previously computed. This indicator is used to apply a cost to the roads. The algorithm will find the shortest path with the lowest cost. A low cost is represented by a high indicator value, so the algorithm will find a route that maximizes the indicator values.

Once the computation is made, a line is represented on the map displaying the route between the two locations (Figure 4.10 for the desktop application and Figure 4.11 for the mobile application).

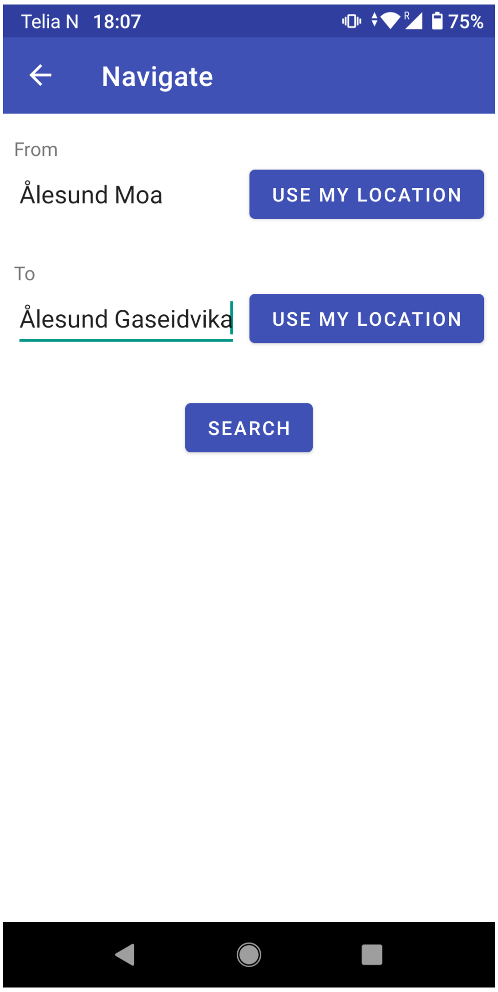


Figure 4.9: Navigation screen (mobile application).

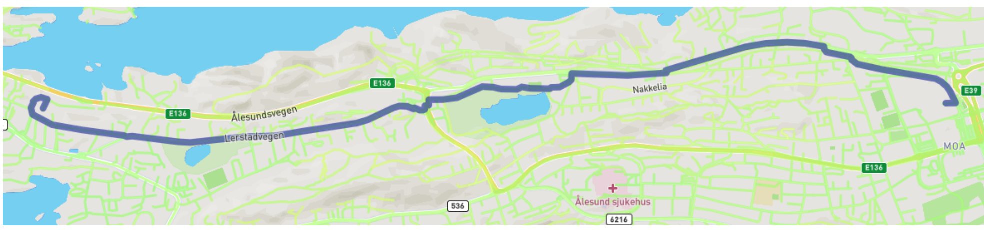


Figure 4.10: Navigation result (desktop application).

Figure 4.11: Navigation result (mobile application).

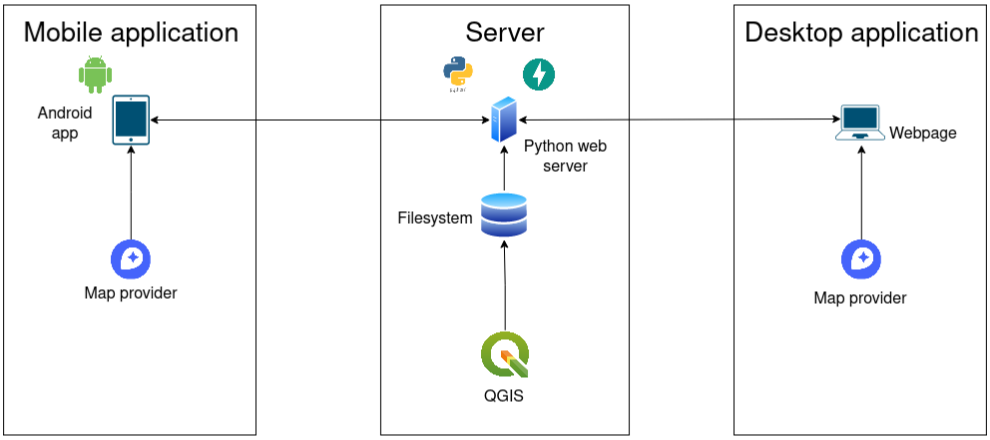


Figure 4.12: Implementation diagram.

4.2.3 Implementation aspects

Figure 4.12 shows the implementation diagram of the project.

Server

The server consists of three parts: pre-processing, storage, processing, and handling of web requests. The pre-processing part was done using QGIS.[1] QGIS is a free and open-source GIS software. It was used to create the raw indicator values file. Two pieces of information are stored: the indicator values (stored in a vector data file) and some additional information about the indicators, such as their names or description (stored in a JSON file).

As for GENOR, the server had to be a REST API that can work with GIS. Therefore, Python was used with the FastAPI package.

Mobile application

Due to the availability of devices, the choice has been made to develop an Android application. The minimum API level required to run the application is 16 (corresponds to Android 4.1), which represents 99.8% of Android devices (as of March 2022). The Android SDK was used to develop the application. It is a collection of software development tools and libraries required to develop Android applications and developed by Google. This is the regular way of developing Android applications. The programming language used is Kotlin.

As for GENOR, the Mapbox map provider was used to display the map in the application.

[1] `https://qgis.org/en/site/` (As of May 2022).

Desktop application

The desktop application is a website. This choice was made because a website can be visualized on any device with any screen size. Moreover, the already implemented server can be used to serve the webpage.

A webpage consists of HTML, CSS, and JavaScript code. The Bootstrap framework[2] was used. It is a free and open-source tool which can be used to have a responsive website, meaning it is suited for any screen, from very small to very large. As for the mobile application, the Mapbox map provider was used to display the map.

4.3 On the visualization of walkability and accessibility

A practical use case of the two applications was conducted on walkability and accessibility. The following indicators were used:

- Elevation: A place is considered more walkable if the elevation is low.
- Population density: A higher density means a more walkable area.
- Proximity of parks: This indicator is higher when a park is nearby. The maximum distance from which a park is too far can be defined by the user.
- Street connectivity: The more street intersections, the better for walkability.
- Speed limit: It is safer to walk when the car speed limit is low.
- Pedestrian crossings: This indicator is higher when pedestrian crossings are nearby. The maximum distance from which a pedestrian crossing is too far can be defined by the user.
- Slope: A road with a high slope is less walkable. The biggest slope considered to be walkable can be defined by the user.
- Distance to schools: This indicator is higher when schools are nearby. The maximum distance from which a school is too far can be defined by the user.
- Distance to health facilities: This indicator is higher when health facilities such as hospitals or nursing home are nearby. The maximum distance from which a health facility is too far can be defined by the user.
- Distance to lights: This indicator is higher when public lights are nearby. The maximum distance from which a public light is too far can be defined by the user.

4.3.1 Dataset used

Global and local datasets have been used to compute indicator values.

The population dataset[3] gives the population density with 250m squares. The locations of parks, schools and health facilities were obtained from the property re-

[2] `https://getbootstrap.com/` (As of May 2022).

[3] `https://data.geonorge.no/sosi/befolkning`, dataset ID: befolkning_rutenett_250_2019 (As of June 2022)

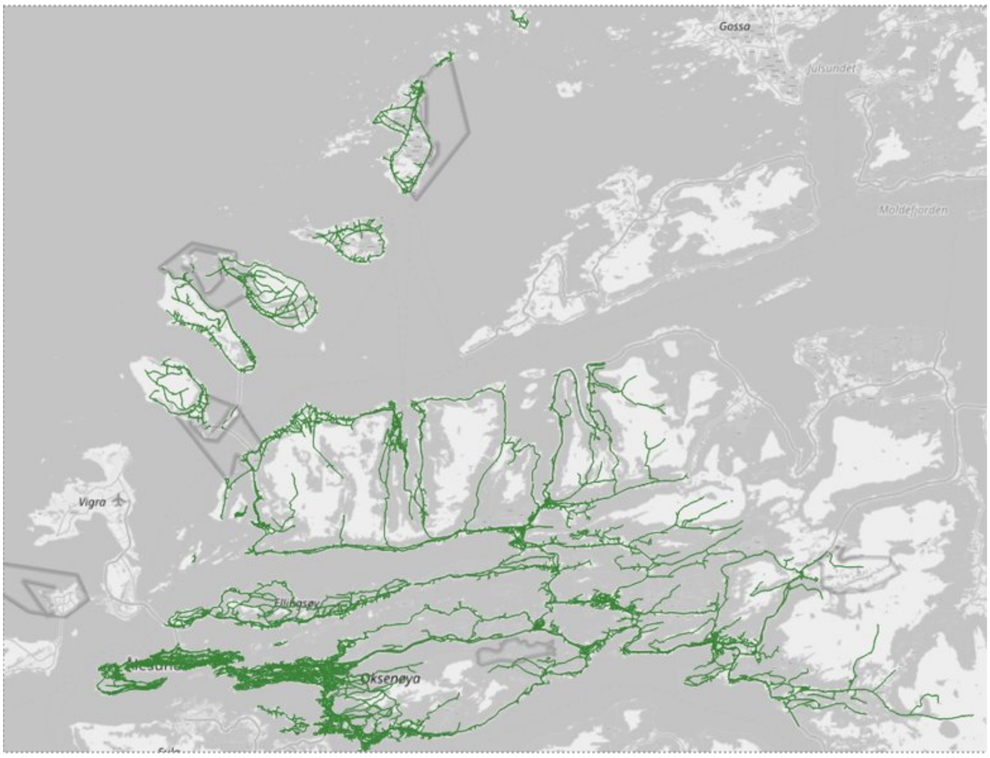


Figure 4.13: Area of study.

gister[4]. Public lights were obtained from the common database of Norway (FKB)[5], as well as an elevation model used to compute the elevation and slope indicators. Finally, the pedestrian crossings, number of street intersections, speed limit come from OpenStreetMap,[6] as described in the GENOR chapter.

4.3.2 Indicator definition

The indicators were computed for Ålesund and its surroundings. Walkability and accessibility are linked to mobility, so the indicators values were only computed on the roads located in the area of interest. Figure 4.13 shows the area of study, as well as the location of where the indicators were computed.

As described in the implementation section, the QGIS software was used to compute the indicators. For each indicator, the goal was to assign a raw indicator value to each road located in the area of interest.

[4] `https://data.geonorge.no/sosi/matrikkel`, dataset ID: bygningspunkt (As of June 2022)

[5] `https://www.kartverket.no/geodataarbeid/geovekst/fkb-produktspesifikasjoner` (As of June 2022).

[6] `https://www.openstreetmap.org/` (As of May 2022).

Population density

The population dataset was giving the number of people living in 250m squares. Therefore, it consisted of vector data with population values associated with polygons.

The values are associated with some polygons, but the goal is to have values associated with any location within the area of interest. Therefore, an interpolation is needed. QGIS proposes the Triangulated irregular network (TIN) interpolation:[7] it creates a network of non-overlapping triangles, and then a value at any location is computed as a weighing of the values of the three apexes of the triangle to which it belongs. This result in a raster layer, consisting of rows and columns of pixels, each one containing a value. Finally, a population value is given to each road by taking the corresponding raster value. This gives the raw population indicator value. Equation 3.2 was used to compute the indicator value.

Proximity of parks

The parks dataset consisted of polygons, each one being a park. This indicator is related to the distance between a point and the nearest park, so a distance function from QGIS was used to compute the raw values.

Equation 4.1 was used to compute the indicator value. The $maximalDistance$ variable can be given by the user. By default, it is 800 meters (the farthest radial distance based on a ten minutes walk).

$$distanceToParkIndicator = \max(1 - \frac{distanceToPark}{maximalDistance}, 0) \tag{4.1}$$

Pedestrian crossings

The pedestrian crossings dataset consisted of points, each one being a crossing. The same method as for the proximity of parks indicator was used.

Street connectivity

The street intersection dataset consisted of points, each one being an intersection. This indicator is related to the number of street intersections within a certain space. For each location, the number of intersections within 800 meters is counted. This gives the raw street connectivity indicator value. Equation 3.4 was used to compute the indicator value.

Elevation

The elevation dataset was a raster layer containing elevation values for the whole region of Ålesund. Therefore, the elevation of to each road is taken from the corresponding raster value. Equation 3.5 was used to compute the indicator value.

[7] `https://en.wikipedia.org/wiki/Triangulated_irregular_network` (As of May 2022).

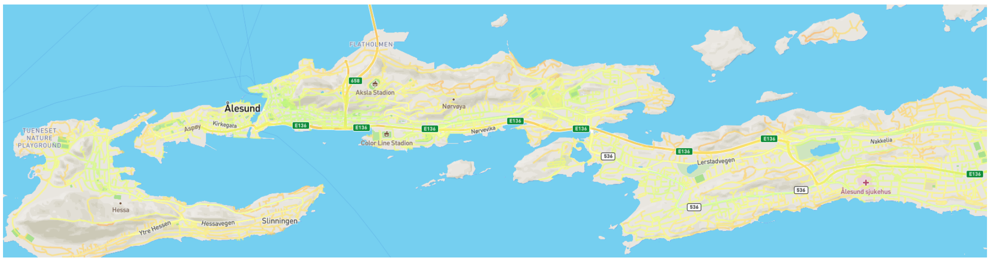


Figure 4.14: Visualization of walkability and accessibility in Ålesund.

Figure 4.15: Visualization of the distance to schools indicator in Ålesund.

Slope

The slope dataset was also a raster layer, so the same method as the elevation indicator was used. However, the user can define define the biggest slope considered to be walkable, as shown in Equation 4.2 with the $maximalSlope$ variable.

$$slopeIndicator = \max(1 - \frac{slope}{maximalSlope}, 0) \tag{4.2}$$

Distance to schools / health facilities / lights

The datasets related to these indicators consisted of polygons, so the same method as for the proximity of parks indicator was used.

4.3.3 Walkability and accessibility visualization

Figure 4.14 shows the result with all indicators selected and all weights set to 1 for Ålesund. We can see that the city center is the most walkable part of the city, and that walkability and accessibility are overall not high near the sea.

The tool can also be used to visualize single indicators. For example, Figure 4.15 shows only the distance to schools indicator. The red roads indicate locations which are at least 800 meters away from a school.

Chapter 5

Evaluation

This chapter presents the evaluation for the mobile and the desktop applications described in Chapter 4. The purpose of the validation was to determine if the applications are relevant and useful for the target users. Two evaluations were conducted: one on the desktop application, and the other on the mobile application presented in Chapter 4.

5.1 Protocol used

The validation was done through empirical evaluation using the DECIDE protocol, which has previously been adopted in several evaluations [33]. This protocol consists of six different steps:

- Determine the goals.
- Explore the questions by defining the questions to be answered by the evaluators.
- Choose the evaluation paradigm and techniques according to the actual situation of the evaluation.
- Identify the practical issues such as users, equipment, materials, schedules, and budgets.
- Decide how to deal with the ethical issues by protecting people's privacy for example.
- Evaluate, interpret, and present the data.

5.2 Desktop application evaluation

This section presents an evaluation carried on the desktop application described in Chapter 4.

5.2.1 DECIDE protocol

This section presents the six steps of the DECIDE protocol applied to the evaluation of the desktop application.

Determine the goals

The goal was to assess the graphical user interface of the application and the usability of the different features. Users were expected to easily understand how the application works and to find how the application can be used in urban planning.

Explore the questions

The following research questions were expected to be answered by the user study:

- How easy is it to understand and use the application?
- Do the results displayed by the application are consistent with real life?
- Is the application relevant for urban planning?

Choose the evaluation paradigm and techniques

The user study was conducted on a printed form consisting of selection options, scale ratings, and free text fields. The results were reported on an online sheet.

Identify the practical issues

- Users: A group of people attending the North West[1] conference in Ålesund, an event about sustainable regional development.
- Equipment: Each user interacted with the application through a laptop. Then, they were asked to fill a written questionnaire.
- Materials:
    - Instructions were given orally on how to use the application, as well as background information about this work.
    - A webpage to access the application.
    - A form, divided into a profile part and an application part.

Decide how to deal with the ethical issues

The responses to the form were collected anonymously, unless the participant was filling to share his/her email address.

Evaluate, interpret, and present the data

The responses of the form were collected using selection options, scale ratings, and free text fields, which were then used to create different types of charts.

[1] `https://www.thenorthwest.no/en` (As of May 2022).

5.2.2 Evaluation form

The evaluation form is divided into three parts: profile part, application part, and conclusion part.

Profile part

The goal of the profile part was to obtain general information about the participants. The first two questions referred to the level of education and the professional activity of the participants (between public sector, private sector, and student). The other three questions focused on assessing the familiarity of participants with different concepts related with the application. The following questions were utilised:

- How do you assess your knowledge about Ålesund?
- How familiar are you with map-based applications (e.g. Google Map, Bing Map, Mazemap, etc.)?
- How familiar are you with geographic information system (GIS)?
- How familiar are you with information visualization approaches?
- How familiar are you with the walkability concept?

On each question, the participants had the choice among:

- Not at all familiar
- Somewhat familiar
- Familiar
- Very Familiar
- Completely familiar

Another question was related to the definition of walkability. Participants were asked to choose one or more indicators that best describe walkability among the following:

- Elevation
- Slope
- Population density (number of people living in the area)
- Car speed limit
- Street connectivity (number of street intersections)
- Proximity of pedestrian crossings
- Distance to schools
- Distance to parks
- Distance to health facilities
- Distance to good illuminated sidewalks or areas

Participants also had the possibility to suggest other indicators.

Application part

Before filling out this part, participants had to use the application by selecting indicators, assigning weights and visualizing results. Then, the first questions of this part aimed at assessing the ease of use of the application:

- How easy is it to understand the definition of indicators?
- How easy is the selection of indicators?
- How easy is it to set the values of indicator parameters?
- How easy is the assignment of weights for each indicator?
- How easy is the identification of roads with low and high values of walkability?
- Overall, how do you rate the degree of difficulty of using the application to estimate walkability (selecting indicators and assigning the weights)?

On each question, the participants had the choice among:

- Very difficult
- Difficult
- Medium
- Easy
- Very easy

Another question referred to whether the participant agreed with the results presented by the application, among the following choices:

- Strongly disagree
- Disagree
- Neutral
- Agree
- Strongly agree

Conclusion part

The conclusion was focused on the understanding of the application and its relevancy for urban planning. The first question asked participants to scale their understanding of the application between 1 (no understanding) and 5 (complete understanding). The second question asked if the application is relevant for urban planning by choosing among:

- Strongly disagree
- Disagree
- Neutral
- Agree
- Strongly agree

Finally, participants had the possibility to leave free comments related to the use of the envisioned application.

5.2.3 Results for the evaluation of the desktop application

Seven people attended the evaluation.

Profile part

The majority of participants have a master's degree, while the rest has a bachelor's (Figure 5.1a). Half of the participants were students, some of them work in the public sector and the rest work in the private sector (Figure 5.1b).

Most of the participants were familiar or very familiar with map-based applications (Figure 5.1c). Also, most of the participants were familiar with GIS (Figure 5.1d). However, participants were less familiar with the concept of information visualization approach (Figure 5.1e).

The assessment of the participants' knowledge of Ålesund were heterogeneous (Figure 5.1f). About the familiarity with the concept of walkability, almost half of the participants were somewhat familiar, and the other half were completely familiar (Figure 5.1g).

The last question of the profile part asked which indicators are relevant in the definition of walkability (Figure 5.1h). The most relevant indicators was speed limit, while the elevation and distance to health facilities were not considered relevant.

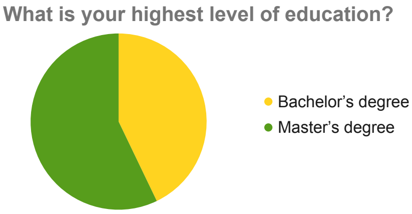
The results of the profile part of the questionnaire indicate several pieces of information.

First, most of the participants are students with a strong academic background. The desktop application evaluated in this part is aimed to be used by urban planners. Since half of the participants are students, the results of this questionnaire may not exactly reflect the view of urban planners. However, this evaluation allows a first assessment of the visual aspect of the application.

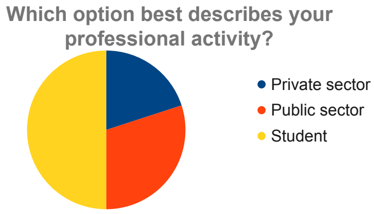
Second, not all participants were familiar with Ålesund. This means that for the question asking whether the participant agree with the results presented by the application, the results of this question are only relevant for those who are familiar with Ålesund.

Third, on average, participants were completely familiar with map-based applications, very familiar with GIS and familiar with information visualization approaches. Therefore, it can be assumed that they have sufficient knowledge to be able to use and assess the application.

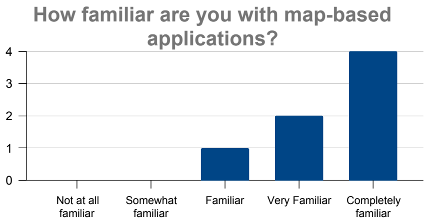
Fourth, all indicators were designated relevant by at least half of the participants except for the distance to health facilities and elevation indicators. As most of the participants were students, it is understandable that they do not need access to health facilities, so this indicator might still be relevant for other groups. However, the elevation indicator does not seem to be relevant and thus should be removed from the application.



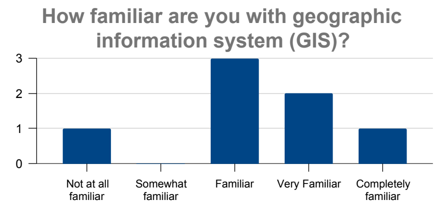
(a) Level of education.



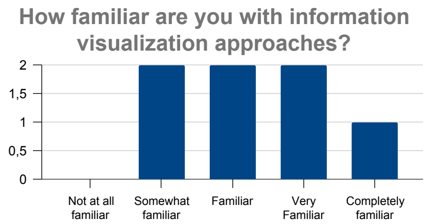
(b) Professional activity.



(c) Knowledge about map-based applications.

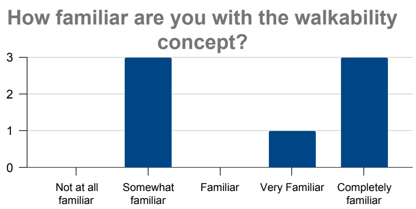


(d) Knowledge about GIS.

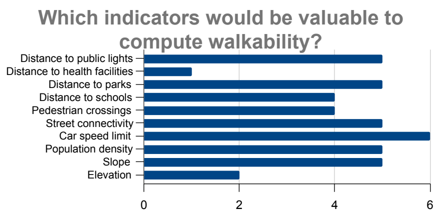


(e) Familiarity with information visualization approaches.

(f) Knowledge about Ålesund.



(g) Familiarity with the walkability concept.



(h) Most relevant indicators for walkability.

Figure 5.1: Results of the profile part.

Application part

The first questions of this part were about indicators: understanding of the definition of indicators (Figure 5.2a), easiness of selecting indicators (Figure 5.2b), easiness to set the values of indicator parameters (Figure 5.2c), and easiness of assigning weights to each indicator (Figure 5.2d). All but one participant indicated that these four tasks were easy or very easy.

The task of identifying roads with low and high values of walkability (Figure 5.2e) was more difficult for the participants, as the average result was between medium and easy. Overall, the participants found that the application was easy to use (Figure 5.2f). Finally, all participants agreed with the results presented by the application (Figure 5.2g).

The results of this part show that the application is overall easy to use and presents results that are coherent with real life. However, the visualization of roads with low and high values of walkability should be improved, as it was considered difficult by a significant part of the participants.

Conclusion part

The conclusion focused on the overall understanding of the application and the relevancy of it for urban planning. The results (Figures 5.3a and 5.3b) show that participants understood how the application works and find it relevant for urban planning. However, as most of the participants were students, the relevancy of the application for urban planning could not be entirely proven with this evaluation. Another evaluation with real urban planners would be needed to address this.

5.3 Mobile application evaluation

This section presents an evaluation carried on the mobile application described in Chapter 4.

5.3.1 DECIDE protocol

This section explains the details of the six steps of the DECIDE protocol applied to the evaluation of the mobile application.

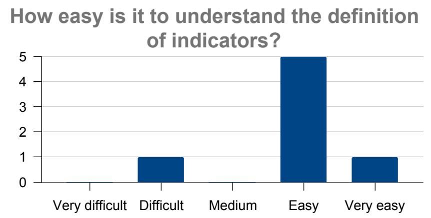
Determine the goals

The goal of this study was to assess the graphical user interface of the application, as well as the usability of the different features. Users were expected to easily understand how the application works and to find the application useful for urban planning.

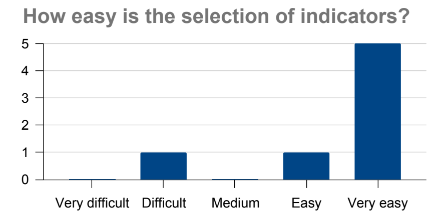
Explore the questions

The following research questions were expected to be answered by the user study:

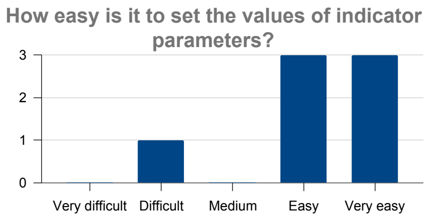
- How easy is it to understand and use the application?
- Do the results displayed by the application are consistent with real life?



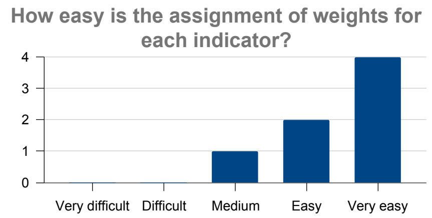
(a) Understanding of the definition of indicators.



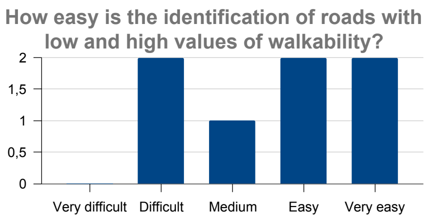
(b) Easiness of the selection of indicators.



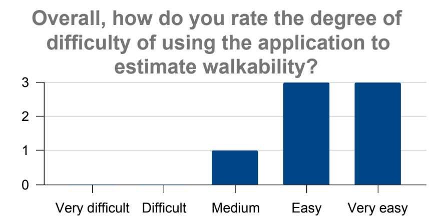
(c) Easiness of the setting of indicator parameters.



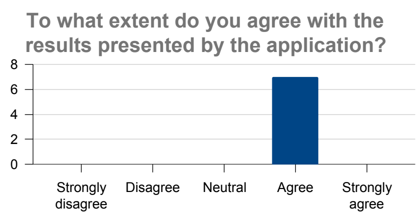
(d) Easiness of the assignment of weights.



(e) Easiness of the identification of roads with low and high values of walkability.



(f) Degree of difficulty of using the application.

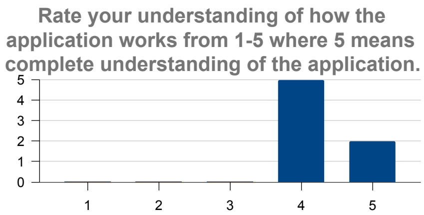


(g) Coherence of the application with real life.

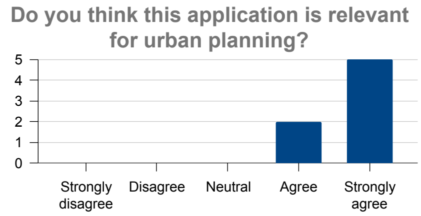
Figure 5.2: Results of the application part.

Choose the evaluation paradigm and techniques

The user study was conducted on a printed form consisting of selection options, scale ratings, and free text fields. The results were then reported on an online



(a) Understanding of the application.



(b) Relevancy of the application for urban planning.

Figure 5.3: Results of the conclusion part.

sheet.

Identify the practical issues

- Users: A group of workers of the Nordvest Fiber company.[2]
- Equipment: Each user was interacting with the application through their mobile. Then, they were asked to fill a written questionnaire.
- Materials:
  - Instructions on how to use the application, as well as some background information about this work, were given orally.
  - A link to download the mobile application.
  - A form, divided into a profile part and an application part.

Decide how to deal with the ethical issues

The responses to the form were collected anonymously.

Evaluate, interpret, and present the data

The responses of the form were collected using selection options, scale ratings, and free text fields, which were then used to create pie charts, bar charts, and column charts.

5.3.2 Evaluation form

The evaluation form is divided into three parts: profile part, application part, and conclusion part.

Profile part

The goal of the profile part was to obtain general information about the participants.

[2] `https://www.nordvestfiber.no/kontaktoss/` (As of May 2022).

The first question referred to the level of education of the participants. The second question concerned assessing the familiarity of participants with with map-based applications. The participants had the choice among:

- Not at all familiar
- Somewhat familiar
- Familiar
- Very Familiar
- Completely familiar

The next two questions were about the use of mobile applications. The first question asked the participants if they use mobile applications, while the second question asked the frequency of their use of applications that help them in everyday life. For this last question, participants had the choice among:

- No
- Barely
- Yes, several times a month
- Yes, several times a week
- Yes, several times a day

The last question of the profile asked the participants if they use applications specifically made for handicaped people.

Application part

Before filling out this part, participants had to use the application to select indicators, assign weights, visualize the results and save the indicators. Then, the first questions of this part concerned assessing the ease of use of the application:

- How easy is it to understand the definition of indicators?
- How easy is the selection of indicators?
- How easy is it to set the values of indicator parameters?
- How easy is the assignment of weights for each indicator?
- How easy is it to save the indicators?
- How easy is the identification of routes with low and high values of mobility?
- Overall, how do you rate the degree of difficulty of using the application to estimate and visualize mobility (selecting indicators and assigning the weights)?

On each question, the participants had the choice among:

- Very difficult
- Difficult
- Medium
- Easy
- Very easy

Another question asked whether the participant agreed with the results presented by the application, among the following choices:

- Strongly disagree
- Disagree
- Neutral
- Agree
- Strongly agree

The final question was about the relevancy of indicators for walking and bicycling. For each indicator, the participants could indicate if it is not relevant, some relevant, or very relevant. The considered indicators are:

- Slope
- Road speed limit
- Pedestrian crossings
- Distance to light
- Population density
- Distance to school
- Proximity of parks
- Distance to health facilities

Conclusion part

The conclusion was focused on the understanding of the application. The first question asked participants to scale their understanding of the application between 1 (no understanding) and 5 (complete understanding). The second question referred to giving participants the opportunity to provide a free comment related to the use of the envisioned application.

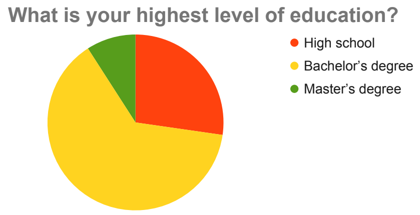
5.3.3 Results for the evaluation of the mobile application

Eleven people attended the evaluation. This section presents and then discuss the results of the form.

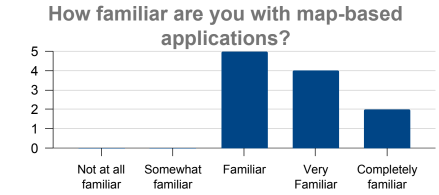
Profile part

The majority of participants have a bachelor's degree, some stopped studying after high school and one has a master's degree (Figure 5.4a). The participants were on average very familiar with map-based applications (Figure 5.4b). Participants used mobile applications at least several times a week (Figure 5.4c and Figure 5.4d), but nobody used applications designed for people with reduced mobility (Figure 5.4e).

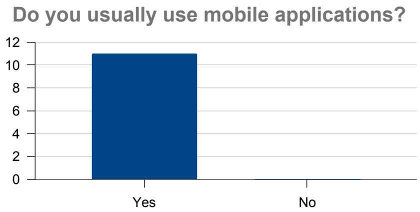
The results of this part show that participants were very familiar with map-based mobile applications. This means that they had sufficient knowledge to use and assess the application. However, no participant had reduced mobility, which is one of the target group of the mobile application. Therefore, the usefulness of the application for people with reduced mobility was not be assessed.



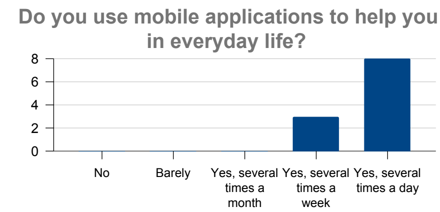
(a) Level of education.



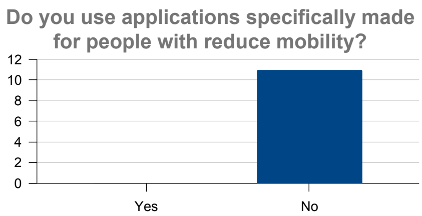
(b) Knowledge about map-based applications.



(c) Usage of mobile applications.



(d) Frequency of using mobile applications.



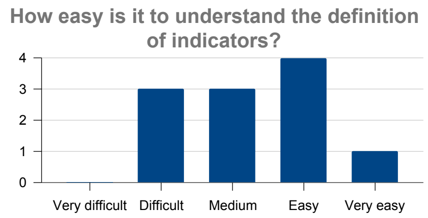
(e) Usage of mobile applications designed for people with reduced mobility.

Figure 5.4: Results of the profile part.

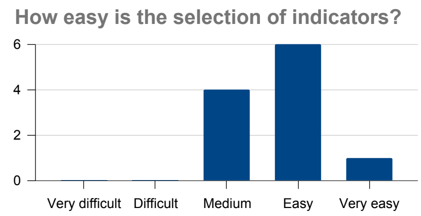
Application part

The first questions of this part were about indicators: understanding of the definition of indicators (Figure 5.5a), easiness of selecting indicators (Figure 5.5b), easiness of setting the values of indicator parameters (Figure 5.5c), easiness of assigning weights to each indicator (Figure 5.5d), and easiness of saving indicators (Figure 5.5e). The selection and saving of indicators was on average easy. However, the definition of indicators, setting the values of indicator parameters and assigning the weights were more difficult.

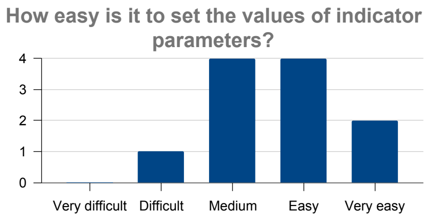
The difficulty of identifying roads with low and high values of mobility (Figure 5.6a) was on average described as medium by the participants. The overall difficulty of using the application (Figure 5.6b) is also described as medium, but on average participants agreed with the results presented by the application (Fig-



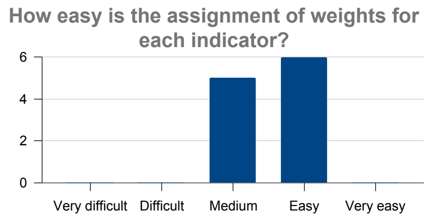
(a) Understanding of the definition of indicators.



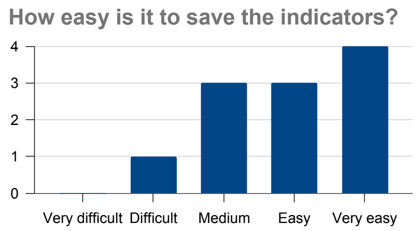
(b) Easiness of selecting indicators.



(c) Easiness of setting indicator parameters.



(d) Easiness of setting indicator weights.

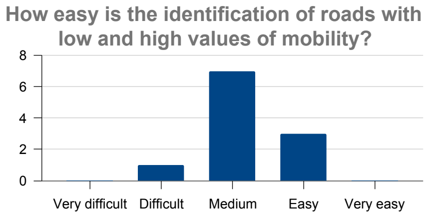


(e) Easiness of saving indicators.

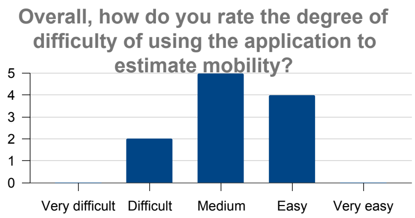
Figure 5.5: Results of the application part (part 1).

ure 5.6c). The two last questions of this part were about the relevancy of indicators when walking and bicycling. Figure 5.6d presents the results when walking. The slope and proximity to parks were the most relevant indicators, followed by the road speed limit, the distance to lights, and the population density indicators. The proximity of pedestrian crossings, schools and health facilities were not considered relevant by the majority of participants. Figure 5.6e presents the results in the case of bicycling. Here, the slope, road speed limit and distance to lights were the most relevant indicators, while the proximity of pedestrian crossings, population density, and proximity of parks only had a medium relevance. The distance to schools and health facilities were not considered relevant.

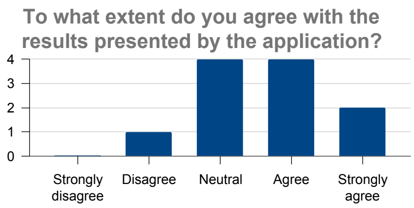
The results of this part show multiple pieces of information. First, the selection and saving of indicators were considered easy to use, so do not require changes. However, the results show that setting the values of indicator parameters and



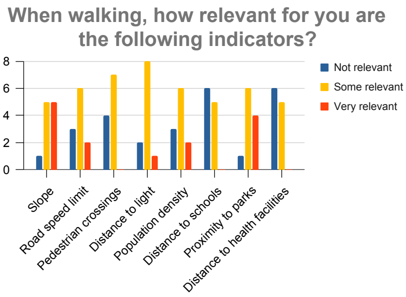
(a) Easiness of the identification of roads with low and high values of mobility.



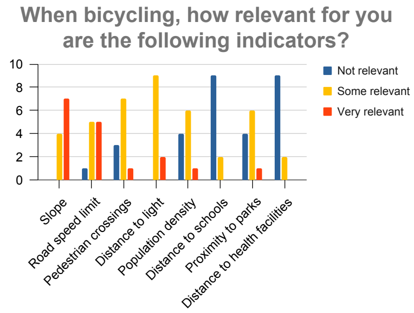
(b) Degree of difficulty of using the application.



(c) Coherence of the application with real life.



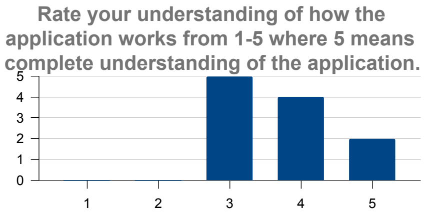
(d) Most relevant indicators for walking.



(e) Most relevant indicators for bicycling.

Figure 5.6: Results of the application part (part 2).

assigning weights should be made in a more intuitive way. Moreover, the definition of each indicator could also be improved. Second, participants had difficulties to identify roads with low and high values of mobility. Therefore, this part should be made in clearer way. Third, the proximity of schools and health facilities were not considered relevant by the participants for both walking and bicycling. As this study group did not include anyone with reduced mobility, it can be considered that these indicators might still be relevant for other users.



(a) Understanding of the application.

Figure 5.7: Results of the conclusion part.

Conclusion part

Figure 5.7a presents the overall understanding of the application. Overall, the participants understood how the application works.

The last question asked participants provide some comments related to the application. Here is a list of what they wrote:

- There should be a default setting for bicycling, walking, wheelchair, etc. This was written by several participants.
- Use less text and more icons.
- There are too many choices.
- Let choose between show or hide information on the select indicators menu.
- Make the legend more informative.
- Use a different color scale for the visualization of mobility.

These comments can be classified into two parts: adding feature (with the first comment) and improving the visual aspect of the application (with the other comments). Addressing these comments may improve the overall usability of the application.

Chapter 6

Discussion and Conclusions

This chapter presents a discussion about the tools developed during this thesis in Section 6.1, and a conclusion is made in Section 6.2. Possible directions for future work are presented along with the discussion.

6.1 Discussion

The visualization tools for spatial data constructed in this thesis are in their first stage of development and thus they need further research to be implemented properly. However, the performed validation showed that these tools have the potential to improve the understanding and communication of spatial data.

6.1.1 Computer Science aspects

A comparison between GENOR and the mobile and desktop applications introduced in Chapter 4 is presented in Table 6.1. The applications have similar architecture, but the implementation of the client application and the data storage system are different.

Both tools use a client-server architecture, which means there would be one server and multiple clients. This has multiple advantages. For example, the data processing is done on the server, so only the server has to be computationally effective and thus the client can run on a low-end device without any effect on the speed of computation. Moreover, data are only stored on the server and not on every client, reducing the size of each client application and allowing collaboration between clients.

On the one hand, GENOR relies on Unity, which is a convenient tool to create 3D visualizations. However, Unity applications are programs, which means they need to be compiled for each target operating system. For example, if we want GENOR to be accessible on any computer, it must be compiled for Linux, Windows, and macOS. If other devices are used (such as mobile phones, or single-board computers with a different Central Processing Unit (CPU) architecture like the Raspberry Pi), GENOR must also be compiled for these devices. Furthermore,

Table 6.1: Comparison of GENOR with the mobile and desktop applications of Chapter 4.

	GENOR	Mobile/Desktop application
Architecture	Client - Server	Client - Server
Client application	Unity	Website / Android application
Server application	Python web server (REST API)	Python web server (REST API)
Data storage system	Database	File system
Indicators and area of interest	Any	Any
Computation of indicators	User has to provide a file	Pre-computed and stored
Views	2D and 3D	2D
Visualization of indicators	Tiles (2D) / Bars (3D)	Along roads
Visualization space	Raster output	Vector output

after each update of GENOR, the compilation must happen again for each target operating system and device. On the other hand, the mobile and desktop applications of Chapter 4 are a website and an Android application. They have less 3D capabilities, but their deployment is easier. Indeed, a website can be accessed with a web browser on any device with any operating system, and an Android application can be easily deployed for Google Play.[1] These choices of using Unity or a website / Android application can be explained by the target user of each application. GENOR was meant to only be used by urban planners who use computers and want to have 3D visualizations. The applications developed in Chapter 4 were made for both urban planners and citizens, hence the development of a website and a mobile application that were easily to deploy on any device.

A Python web server is used for both tools. This choice has been made because of the easiness of using Python for handling GIS data and creating a server. A technical limitation of the server occurred during the evaluation of the mobile application. Computing the indicators took a lot of time when eleven participants were using the application simultaneously. This can be explained by the quality of the server, which had low hardware specifications. Improvements on the efficiency of the algorithms can also help to make the application faster and easier to use.

GENOR uses a database as a data storage system, while the mobile and desktop applications use a file system. Databases are more complex to design and understand, but they are more flexible, secure, and efficient than file systems. The choice

[1] `https://play.google.com` (As of May 2022).

has been made to have different data storage systems because of how data are managed in the developed tools. For the mobile and desktop applications, data are pre-computed and stored into two different files. Therefore, the data storage system is very simple and did not require the complexity of a database. GENOR lets the user define, edit, and delete data. These operations are complex and have to be executed fast, hence the use of a database.

Both tools are generic, because they can integrate others indicators and they can apply in other areas of interest. However, the computation of indicators and the visualization differ. GENOR presents more visualization methods and the indicator can be defined by the user, so this tool has more features but is more difficult to use. As stated before, one of the target group of the applications developed in Chapter 4 is citizens, who want to have an easy and simple application. Therefore, the desktop and mobile applications were designed in a way to be as simple as possible, so that anybody can use them without any previous knowledge.

6.1.2 Validation

The evaluations presented in Chapter 5 showed the strengths and the weaknesses of the applications. Overall, the applications were easy to use and relevant for urban planning, but some aspects could be improved. For example, the participants said that the mobile application should be made more user-friendly by reducing the number of choices. This could be done by defining default settings depending on the condition of the user, and by displaying less text on the application. Another problem identified by the evaluations was about the color scale used by the visualization of indicators. Some participants said that the colors used were too similar. Therefore, further research is need it in the user-interface applications.

One of the target group of the mobile application presented in Chapter 4 was people with reduced mobility and elderly people. However, no evaluation has been made with them and they might have different remarks than the participants who attended the evaluations. Therefore, conducting a user study with this group of people would be very valuable to assess the mobile application.

The evaluation of the mobile application used the DECIDE framework, which applies to any visualization structures. One way of improving this evaluation would be to use a protocol specifically made to address the interface of mobile applications. For example, Nicastro et al. [34] proposed a guideline composed of 27 semiotic-informed rules for interface evaluation, divided into physical world, empiric, syntactic, semantic, pragmatic, and social world steps. This structure would be beneficial for the assessment of the mobile application since it would help the identification of good and negative features of its design.

6.1.3 Urban planning aspects

Indicators used in land planning were used as an example of visualization tools. The main goal was to create a tool to improve communication of data and not to develop data analysis by itself. GIS are with no doubt the main tools used for the management and analysis of spatial data due to their large number of designed tools and permanent development. Nevertheless, GIS requires a large amount of training and knowledge in spatial statistics which might limit their use. The visualization tools developed here do not intend to replace the capabilities of GIS but rather to facilitate the communication of spatial data by allowing an easy presentation of geodata and the estimation of indicators that can be displayed in 2D or 3D scenes.

Walkability and accessibility indicators were used as an example of visualization tools. Despite the positive results of the validation, some limitations were observed in the calculation of the indices and thus some improvements are needed. In the actual version, the standardization assumes that variables have a linear relation with the estimated indicators. These problems can be resolved by including a more flexible algorithm that allows to predetermine maximum and minimum values, use different functions of relationships. However, increasing the capability of the algorithm might also increase the complexity of using the tool and thus works against the usability of the tool.

6.2 Conclusions

As more and more urban data become available, visualization tools are needed to support the proper understanding and analysis of such data. This thesis proposed the design and implementation of three visualization tools: GENOR, a generic platform for indicator assessment, and a generic platform for indicator visualization for mobile and desktop environments. Two case studies were conducted to evaluate the applications. The four research questions of this thesis are answered:

- **RQ1:** How to design a generic platform that supports both the visualization and the assessment of indicators based on spatial data?
  In Chapter 3, the architecture of a generic platform for indicator assessment was presented. The proposed architecture is divided into two parts: a client, to handle the user interaction and visualization, and a server, to handle the data storage and processing. A similar architecture was presented in Chapter 4. The main difference between the two architectures is about data management: Chapter 3 lets the user define or delete indicators, while Chapter 4 uses pre-computed indicators. This choice has been made because of the target users of the applications: GENOR is to be used by urban planners, so it has more features, while the desktop and mobile applications are to be used also citizens, so they mean to be more simple and easy to use.
- **RQ2:** How to implement a generic application that can be easily used by different target users?
  Chapter 3 introduced a generic platform for indicator assessment that can be used by urban planners. The implementation was detailed in this chapter, and relies on the use of free and open-source tools. Chapter 4 introduced a generic platform for indicator visualization that can be used by both urban

planners and citizens. During implementation, emphasis was placed on ease of use for the tool to be accessible to everyone.

- **RQ3:** Would the use of this platform be effective to support the analysis of indicators for different scenarios?
  The different applications were tested on three case studies: bus service availability, walkability, and mobility. They were presented in Chapter 3 and Chapter 4. The implementation of these case studies validated the proposed architectures.
- **RQ4:** Would the developed tools be useful for different user groups?
  Chapter 5 described two evaluations conducted on the applications presented in Chapter 4. Overall, a positive feedback was obtained on the use of the developed tools. Participants said that the applications were easy to use and were relevant for urban planning. Some constructive comments also raised improvements that could be made in a future work.

The results of this research offer several opportunities for future work. The evaluations presented in Chapter 5 provided valuable feedback that would improve the applications. For example, in the mobile application, adding a default setting depending on the condition of the user (e.g. elderly or people with reduced mobility) would make the application easier to use. Likewise, the platform presented in Chapter 3 is simple and adding more features, such as more visualizations methods, would improve the quality of the tool. Another possibility of future work would be about the use of dynamic data. Currently, only static data are used, but adding the possibility to support live data would improve the platforms. For example, the weather conditions would be useful to compute Walkability. Furthermore, performing evaluations with other groups of people would provide more feedback to improve the applications. Finally, the last step would be to publish the applications to make them available for everyone. This would require to employ a dedicated machine powerful enough to host the web server.

Bibliography

[1] Z. Yang and A. Kankanhalli, 'Innovation in Government Services: The Case of Open Data,' in *InternationalWorking Conference on Transfer and Diffusion of IT (TDIT)*, Y. K. Dwivedi, H. Z. Henriksen, D. Wastell and R. De', Eds., ser. Grand Successes and Failures in IT. Public and Private Sectors, Part 7: Shorter Papers, vol. AICT-402, Bangalore, India: Springer, Jun. 2013, pp. 644–651. DOI: `10.1007/978-3-642-38862-0\_47`. [Online]. Available: `https://hal.inria.fr/hal-01467811`.

[2] A. Eberhardt and M. S. Silveira, 'Show me the data! a systematic mapping on open government data visualization,' in *Proceedings of the 19th Annual International Conference on Digital Government Research: Governance in the Data Age*, ser. dg.o '18, Delft, The Netherlands: Association for Computing Machinery, 2018, ISBN: 9781450365260. DOI: `10.1145/3209281.3209337`. [Online]. Available: `https://doi.org/10.1145/3209281.3209337`.

[3] A. Psyllidis, A. Bozzon, S. Bocconi and C. Titos Bolivar, 'A platform for urban analytics and semantic data integration in city planning,' in *Computer-Aided Architectural Design Futures. The Next City - New Technologies and the Future of the Built Environment*, G. Celani, D. M. Sperling and J. M. S. Franco, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2015, pp. 21–36, ISBN: 978-3-662-47386-3.

[4] C. Costa and M. Y. Santos, 'The suscity big data warehousing approach for smart cities,' in *Proceedings of the 21st International Database Engineering & Applications Symposium*, ser. IDEAS 2017, Bristol, United Kingdom: Association for Computing Machinery, 2017, pp. 264–273, ISBN: 9781450352208. DOI: `10.1145/3105831.3105841`. [Online]. Available: `https://doi.org/10.1145/3105831.3105841`.

[5] H. Mehmood, E. Gilman, M. Cortes, P. Kostakos, A. Byrne, K. Valta, S. Tekes and J. Riekki, 'Implementing big data lake for heterogeneous data sources,' in *IEEE 35th International Conference on Data Engineering Workshops (ICDEW)*, 2019, pp. 37–44. DOI: `10.1109/ICDEW.2019.00-37`.

[6] B. Longva, 'Digital twin for walkability assessment in city planning,' M.S. thesis, Norwegian University of Science and Technology (NTNU), 2021.

[7] S. Jones, 'Accessibility measures: A literature review,' *Publication of: Transport and Road Research Laboratory*, no. TRRL LR 967 Monograph, 1981.

[8] C. M. Hall and Y. Ram, 'Walk score® and its potential contribution to the study of active transport and walkability: A critical and systematic review,' *Transportation Research Part D: Transport and Environment*, vol. 61, pp. 310–324, 2018, ISSN: 1361-9209. DOI: `https://doi.org/10.1016/j.trd.2017.12.018`. [Online]. Available: `https://www.sciencedirect.com/science/article/pii/S1361920917302560`.

[9] P. M. Fortini and C. A. Davis, 'Analysis, integration and visualization of urban data from multiple heterogeneous sources,' in *Proceedings of the 1st ACM SIGSPATIAL Workshop on Advances on Resilient and Intelligent Cities*, ser. ARIC'18, Seattle, WA, USA: Association for Computing Machinery, 2018, pp. 17–26, ISBN: 9781450360395. DOI: `10.1145/3284566.3284569`. [Online]. Available: `https://doi.org/10.1145/3284566.3284569`.

[10] J. Perhac, W. Zeng, S. Asada, S. M. Arisona, S. Schubiger, R. Burkhard and B. Klein, 'Urban fusion: Visualizing urban data fused with social feeds via a game engine,' in *2017 21st International Conference Information Visualisation (IV)*, 2017, pp. 312–317. DOI: `10.1109/iV.2017.33`.

[11] L. Leplat, R. da Silva Torres, D. Aspen and A. Amundsen, 'Genor: A generic platform for indicator assessment in city planning,' in *36th International Conference on Modelling and Simulation (ECMS)*, 2022, pp. 245–253.

[12] A. G. O. Yeh, 'The use of gis in urban planning,' in *Geographical Information Systems, Volume 1, Principles and Technical Issues, Second Edition*, P. A. Longley, M. F. Goodchild, D. J. Maguire and D. W. Rhind, Eds., New York, Chichester, Weinheim, Brisbane, Singapore, Toronto: John Wiley & Sons, INC, 1999, ch. 62, pp. 877–888.

[13] M. Schlossberg and E. Shuford, 'Delineating "public" and "participation" in ppgis,' *URISA Journal*, vol. 16, pp. 15–26, Jan. 2005.

[14] R. Kingston, 'Web-based ppgis in the united kingdom,' *Community Participation and Geographic Information Systems*, pp. 101–112, 2002.

[15] J. Huck, J. Whyatt and P. Coulton, 'Spraycan: A ppgis for capturing imprecise notions of place,' *Applied Geography*, vol. 55, pp. 229–237, 2014, ISSN: 0143-6228. DOI: `https://doi.org/10.1016/j.apgeog.2014.09.007`. [Online]. Available: `https://www.sciencedirect.com/science/article/pii/S0143622814002082`.

[16] G. Bugs, C. Granell, O. Fonts, J. Huerta and M. Painho, 'An assessment of public participation gis and web 2.0 technologies in urban planning practice in Canela, Brazil,' *Cities*, vol. 27, no. 3, pp. 172–181, 2010, ISSN: 0264-2751. DOI: `https://doi.org/10.1016/j.cities.2009.11.008`. [Online]. Available: `https://www.sciencedirect.com/science/article/pii/S0264275109001358`.

[17] C. Prado, C. S. y Justicia Ambiental and R. d. C. para el Mejoramiento de las Comunidades, 'Border environmental justice ppgis: Community-based mapping and public participation in eastern tijuana, méxico,' *International Journal of Environmental Research and Public Health*, vol. 18, no. 3, 2021, ISSN: 1660-4601. DOI: 10.3390/ijerph18031349. [Online]. Available: https://www.mdpi.com/1660-4601/18/3/1349.

[18] H. Bagheri and E. Zarghami, 'Assessing the relationship between housing characteristics and children's independent mobility by ppgis method,' *Journal of Housing and the Built Environment*, pp. 1–17, Jan. 2022. DOI: 10.1007/s10901-021-09928-8.

[19] W. Chen, Z. Huang, F. Wu, M. Zhu, H. Guan and R. Maciejewski, 'Vaud: A visual analysis approach for exploring spatio-temporal urban data,' *IEEE Transactions on Visualization & Computer Graphics*, vol. 24, no. 09, pp. 2636–2648, Sep. 2018, ISSN: 1941-0506. DOI: 10.1109/TVCG.2017.2758362.

[20] Z. Feng, H. Li, W. Zeng, S. Yang and H. Qu, 'Topology density map for urban data visualization and analysis,' *IEEE Trans. Vis. Comput. Graph.*, vol. 27, no. 2, pp. 828–838, 2021. DOI: 10.1109/TVCG.2020.3030469. [Online]. Available: https://doi.org/10.1109/TVCG.2020.3030469.

[21] J. P. Bello, C. Silva, O. Nov, R. L. Dubois, A. Arora, J. Salamon, C. Mydlarz and H. Doraiswamy, 'Sonyc: A system for monitoring, analyzing, and mitigating urban noise pollution,' *Commun. ACM*, vol. 62, no. 2, pp. 68–77, Jan. 2019, ISSN: 0001-0782. DOI: 10.1145/3224204. [Online]. Available: https://doi.org/10.1145/3224204.

[22] G. Andrienko, N. Andrienko, W. Chen, R. Maciejewski and Y. Zhao, 'Visual analytics of mobility and transportation: State of the art and further research directions,' *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 8, pp. 2232–2249, Aug. 2017, ISSN: 1558-0016. DOI: 10.1109/TITS.2017.2683539.

[23] R. Kitchin, T. P. Lauriault and G. McArdle, 'Knowing and governing cities through urban indicators, city benchmarking and real-time dashboards,' *Regional Studies, Regional Science*, vol. 2, no. 1, pp. 6–28, 2015. DOI: 10.1080/21681376.2014.983149. eprint: https://doi.org/10.1080/21681376.2014.983149. [Online]. Available: https://doi.org/10.1080/21681376.2014.983149.

[24] C. Prandi, V. Nisi, M. Ribeiro and N. Nunes, 'Sensing and making sense of tourism flows and urban data to foster sustainability awareness: A real-world experience,' *Journal of Big Data*, vol. 8, no. 1, p. 51, Mar. 2021. DOI: 10.1186/s40537-021-00442-w.

[25] E. Sauda, G. Wessel, R. Kosara, R. Chang and W. Ribarsky, 'Legible cities: Focus-dependent multi-resolution visualization of urban relationships,' *IEEE Transactions on Visualization & Computer Graphics*, vol. 13, no. 06,

pp. 1169–1175, Nov. 2007, ISSN: 1941-0506. DOI: 10.1109/TVCG.2007.70574.

[26] T. Johansson, E. Segerstedt, T. Olofsson and M. Jakobsson, 'Revealing social values by 3d city visualization in city transformations,' *Sustainability*, vol. 8, no. 2, 2016, ISSN: 2071-1050. DOI: 10.3390/su8020195. [Online]. Available: https://www.mdpi.com/2071-1050/8/2/195.

[27] J. Döllner and B. Hagedorn, 'Integrating urban gis, cad, and bim data by service-based virtual 3d city models,' in *Urban and Regional Data Management*, CRC Press, 2007, pp. 169–182.

[28] E. Jamei, M. Mortimer, M. Seyedmahmoudian, B. Horan and A. Stojcevski, 'Investigating the role of virtual reality in planning for sustainable smart cities,' *Sustainability*, vol. 9, no. 11, 2017, ISSN: 2071-1050. DOI: 10.3390/su9112006. [Online]. Available: https://www.mdpi.com/2071-1050/9/11/2006.

[29] R. Weinberger and M. N. Sweet, 'Integrating walkability into planning practice,' *Transportation Research Record*, vol. 2322, no. 1, pp. 20–30, 2012. DOI: 10.3141/2322-03. eprint: https://doi.org/10.3141/2322-03. [Online]. Available: https://doi.org/10.3141/2322-03.

[30] R. van Lammeren, J. Houtkamp, S. Colijn, M. Hilferink and A. Bouwman, 'Affective appraisal of 3d land use visualization,' *Computers, Environment and Urban Systems*, vol. 34, no. 6, pp. 465–475, 2010, GeoVisualization and the Digital City, ISSN: 0198-9715. DOI: https://doi.org/10.1016/j.compenvurbsys.2010.07.001. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0198971510000657.

[31] M. Ranzinger and G. Gleixner, 'Gis datasets for 3d urban planning,' *Computers, Environment and Urban Systems*, vol. 21, no. 2, pp. 159–173, 1997, Urban Data Management Symposium, ISSN: 0198-9715. DOI: https://doi.org/10.1016/S0198-9715(97)10005-9. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0198971597100059.

[32] B. M. Wu and J. P. Hine, 'A ptal approach to measuring changes in bus service accessibility,' *Transport Policy*, vol. 10, no. 4, pp. 307–320, 2003, Transport and Social Exclusion, ISSN: 0967-070X. DOI: https://doi.org/10.1016/S0967-070X(03)00053-2. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0967070X03000532.

[33] J. Preece, Y. Rogers and H. Sharp, *Interaction design: beyond human computer interaction*. John Wiley & Sons, INC, 2002.

[34] F. Nicastro, R. Pereira, B. Alberton, L. P. Morellato, M. C. Baranauskas and R. da S. Torres, 'Enterprise information systems (iceis 2015),' in S. Hammoudi, L. Maciaszek, E. Teniente, O. Camp and J. Cordeiro, Eds., ser. Lecture Notes in Business Information Processing. Springer, 2015, vol. 241, ch. Guidelines for evaluating mobile applications: a semiotic-informed approach, pp. 1–26. DOI: 10.1007/978-3-319-29133-826.

NTNU
Kunnskap for en bedre verden