NTNU
Norwegian University of
Science and Technology

Development of a Configurable Well Montage

Managing Industry Complexity and Information Overload

Robert Einarson

Nicolai Fredriksen

Computer Science - Tenth semester
Master's Thesis (TDT4900)
Supervisor: Eric Monteiro
Delivered: 11th June 2019

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

(This page is intentionally left blank.)

Abstract

The ever-growing digitization happens through all industries. The amount of data grows at an exponential rate. This causes information overload, which constitutes a need for new innovative tools for handling data. New concepts like digital twin and BIM aspires to tackle this problem by generating a common data flow across all actors involved. It suggests a future with an interconnected industry that effectively can handle vast amounts of data.

The first part of this master's thesis looks at the future of the oil industry with regard to the concepts mentioned. This with a particular focus on Schlumberger's well planning software; DrillPlan. The second part will look at the case of how to create a *configurable* Well Montage. Well Montage is a DrillPlan feature that generates a one-page summary of the entire drilling program, but users are currently requesting options for customizability. The exploration of this case's domain aims to explain what a one-page summary constitutes, what it functions as, and describe the iterative process of how we made an improved version of Well Montage. The focus of the iterative development was to create as good a user experience as possible based on the principles of user-centered design.

Key Words:
well planning, one-page summary, information overload, UX case, knowledge management, digital twin, user-centered design

Sammendrag

Den stadig økende digitaliseringen skjer gjennom alle bransjer. Mengden data vokser i eksponentiell hastighet. Det medfører en informasjonsmengde som overskrider hva en klarer behandle, noe som utgjør et behov for nye og innovative verktøy for å håndtere data. Nye konsepter som digital tvilling og BIM ønsker å håndtere dette problemet ved å skape en felles dataflyt over alle involverte bransjeaktører. Disse foreslår en fremtid med en sammenkoblet industri som effektivt kan behandle store mengder data.

Den første delen av denne masteroppgaven ser på oljebransjens fremtid med tanke på de nevnte konseptene. Dette med et spesielt søkelys på Schlumbergers brønnplanleggingsprogramvare; DrillPlan. Den andre delen vil se på hvordan man lager en *konfigurerbar* Well Montage. Well Montage er en DrillPlanfunksjon som lager et énsiders sammendrag over hele boreprogrammet, men brukerne ber nå om muligheter for å konfigurere denne. Utforskningen av oppgavens domene tar sikte på å forklare hva et énsiders sammendrag utgjør, hva den fungerer som, og beskrive den iterative prosessen om hvordan vi lagde en forbedret versjon av funksjonen. Fokuset i den iterative utviklingen var å skape en så god som mulig brukeropplevelse, basert på prinsippene om brukersentrert utvikling.

Nøkkelord: brønnplanlegging, énsides sammendrag, informasjonshåndering, brukeropplevelse, kunnskapshåndtering, digital tvilling, brukersentrert utvikling

Acknowledgements

We would first like to thank our supervisor *Eric Monteiro* of the Department of Computer Science at the Norwegian University of Science and Technology (NTNU). Your valuable input, throughout the semester, helped us shape the essence of this project and inspired the completion of this work.

Our gratitude also goes to *Schlumberger Information Solutions Norway Technology Center* in Asker, for offering us this case, and for facilitating the task. You have been more than helpful and we are grateful for the creative freedom you have given us.

Disclaimer

This paper is written for the course TDT4900 - Master's Thesis. The research is a continuation from the previous semesters TDT4501, Specialization Project, which forms the foundation for TDT4900. The resulting master's thesis is fully separated from the TDT4501 project - and should be regarded as strictly distinct work. The master's thesis is written with assistance from Schlumberger SNTC in Asker. The research is independent work, not controlled by any third party actors. It is undertaken by us as students together with our academic supervisor from NTNU. The results are not in any way sponsored by Schlumberger or influenced by any external actors.

We hereby declare that all text written in this publication is original material, not previously published unless otherwise noted. All sources that form the basis for the numerous statements and understandings are cataloged in the references section.

Robert Einarson

Robert Einarson

Nicolai Fredriksen

Nicolai Fredriksen

Contents

List of Figures

List of Tables

Abbreviations

AEC Architecture, Engineering, and Construction.

API Application Programming Interface.

BHA Bottom Hole Assembly.

BIM Building Information Modeling.

SUS System Usability Scale.

UCD User-Centered Design.

UI User Interface.

1 Introduction

As the advancements of technology make it possible to capture more information than ever, the amount of data that is spawned from said technological advances is a matter of course. The abundance of data could lead to the problem of information overload. When dealing with complex systems that are supposed to handle this data, it is still hard to get a clear overview of what data is relevant or not. How do different industries tackle the problem of information overload, and how do they adapt this apparent abundance of data into a benefit? In addition, how do new paradigms and emergent technologies across different industries pose as a source of inspiration and motivation for all? And how does a reluctance towards change pose as a hindrance, when industries have to be more collaborative in order to utilize their individual strengths?

This thesis will look towards the oil industry. They are going through tougher times, and are also in need of a change to fully utilize what the expectations of the 21st century bring forth. The thesis will look at the concept of well planning, which is an advanced and intricate process for planning the drilling of an oil well. It has to be done properly in order to avoid risks with an emphasis on safety, while still maintaining low costs. In well planning too, there is an increasing amount of data and an increasing reliance on the processing and analysis of the data. In some cases, it could be millions of pages of data that has to be made into an appropriate well plan. Thus a need for summarizations of said data could be necessary, as to make the data malleable and facilitate in the creation of knowledge. So how can the oil industry use emergent technologies and be inspired by other industries, in order to stay on top of the problem, and make the great amount of data into an asset instead?

Case Description

To alleviate the intricacies and complexities of well planning, Schlumberger introduced DrillPlan. It is an all-in-one web platform, meaning it includes all the necessary steps of well planning and compiles it into one coherent platform. DrillPlan aids its users by automating repetitive tasks, stores all their data in the cloud. Through this, it enables collaborative features to further enhance the effectiveness of well planning, for the individual companies that are using DrillPlan.

As the well planning comes to a close, the entirety of planning can be processed into what is called a drilling program. It will contain all the data necessary for the drilling of the well to ensue. Depending on the complexity of the well, the drilling program could be several hundred pages long. To solve the triteness of having this large drilling program, Schlumberger introduced, initialized by user feedback, what they call the *Well Montage*. It is a *one-page summary* of the drilling program, meaning; the most important and useful data of the entire well summarized onto a single page.

However, different companies and users may have particular needs from the Well Montage. As it is today, it is a predefined template of data, that is exported into an Excel sheet. Thus, the current Well Montage is too rigid and not compliant with the different needs of the users.

What this project seeks to do is, therefore: create an interactive application that would make it possible for users to configure their own custom Well Montage. With all the data that is already available on DrillPlan, a user-friendly experience is needed for the users to comply with the information overload that may succumb from making a summary of the well plan.[1]

1.1 Research questions

With the project's description in mind, the following are the research questions of which the master's thesis will be based on:

- **RQ1** What challenges lie ahead for the oil industry in terms of complexity management and handling information overload?
  - **RQ1.1** How can DrillPlan's strategy go beyond solely managing the intricacy of well planning?
  - **RQ1.2** How does a one-page summary help in handling information overload?
- **RQ2** How can the current implementation of DrillPlan's Well Montage be enhanced to encompass users' needs?

Thesis Outline

Section 2 gives the relevant background for the rest of this thesis. *Section 3* introduces the case for this thesis, and *Section 4* describes the methods used to reach a viable solution. *Section 5* describes the information gathering that needed to take place in order to lay a solid foundation for what to take into account. *Section 6* documents the paper prototype and the testing of it. *Section 7* documents the second iteration with a non-functional prototype, based on the feedback gained from the paper prototype. *Section 8* documents the last iteration with a functional prototype, which is the main deliverable for Schlumberger. *Section 9* discusses several aspects regarding both the solution that were reached and issues regarding standardization and implementation of new software in a large scale industry. *Section 10* concludes the thesis.

[1]The original master's thesis proposal can be read in its entirety in Appendix A.

2 Literature Study

This section will lay the literary foundation for the rest of the paper by establishing the problem's domain and challenges, but also by exploring similar problems and solutions in other industries.

2.1 Domain Knowledge

To acquire oil there are several steps one has to take before beginning drilling into the subterrestrial. Firstly, one has to perform a preliminary geological search of the area of interest. It requires operations of large amounts of data from several different sources, such as geomagnetic fields, electromagnetic scans, seismic prospecting, and gravitational fields. In addition to this data, one also is comparing and using historical data to establish if the chosen area is applicable for drilling. The geological research is thereby combining historical science with hermeneutic science *[1] (Frodeman **1995**)*. The hermeneutic science is geologists interpreting whether or not oil is present, based on their past experiences and their particular background. As one can never be absolutely certain of the amount of oil that is present, the geology surrounding well drilling, in general, can be seen as *"grey"*, as one interviewed geologist put it: "*Everything is grey in geology and there is very little that is black and white*" *[2] (Almklov and Hepsø **2011**)*.

Secondly, after one has established the presence of oil in a certain location, one has to plan how to advance with creating a well. The process of planning an oil well should result in a well program which has the three following goals: *safety*, *minimum cost* and *usability*. The several steps of well planning are illustrated in Figure 1. Depending on the well site, the complexity may vary e.g. offshore drilling is much more precarious than onshore drilling. Planning a well is cheap compared to drilling and building the well itself, but still, there are those who cut corners here. One should do this carefully to avoid oil leakages and blowouts which may occur from insufficient planning. There is a fine line between planning enough to reach the necessary and desired goal of safety and usability while maintaining the cost to a minimum *[3] (Mitchell, Lake, and Petroleum Engineers **2007**)*.

New Technology

The oil industry is, according to Hubbert's *peak theory*, fast approaching the peak of oil production *[4] (Hubbert* et al. ***1956**)*. The production from conventional deposits[2] have reached its peak in the early 2000s *[5] (Kerschner and Capellán-Pérez **2017**)*. However, the total global oil production has managed to stay at the same level as before thanks to extraction from unconventional sources. This means that even though much of the easy access oil is now extracted, the advances in technology have made it possible to extract oil from oil reservoirs that earlier thought inaccessible. This is part of the reason why we see a yearly increase in the estimated oil reserves left *[6] (Li **2018**)*. To stay on top of the

[2] Conventional oil refers to oil that is extractable from reservoirs using traditional techniques of drilling, pumping, and compression

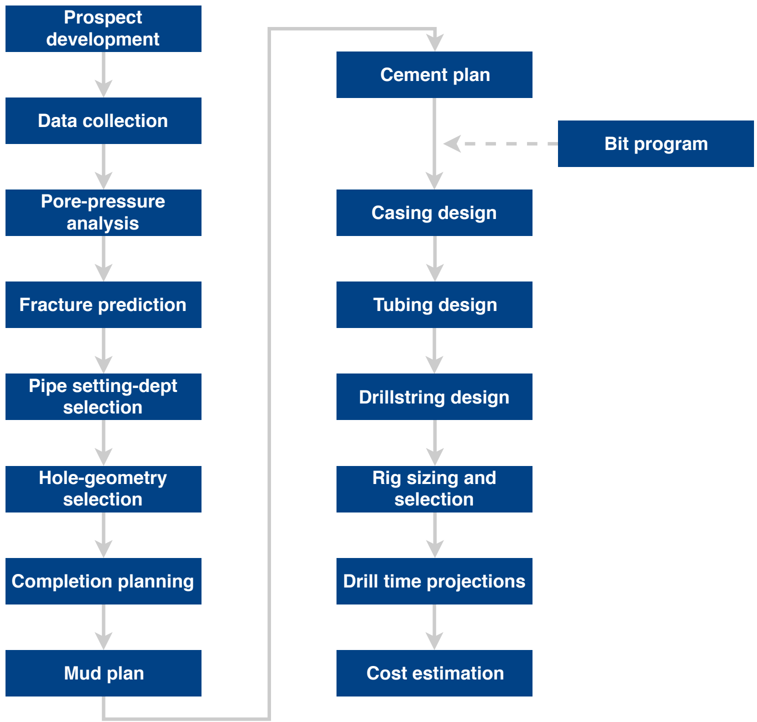


Figure 1: The steps of well planning, based on figure from *[3] (Mitchell, Lake, and Petroleum Engineers **2007**)*.

competition, and in order to stay relevant for both employees and customers, companies have to utilize new technologies effectively *[7] (Bradley, Loucks, Macaulay,* et al. ***2015****)*.

Employment in the oil and gas sector has recently gone through some tough times. The United States Department of Labor reported a drop in employment working in oil and gas extraction from 2015-2017 of almost 29% *[8] (Labor Statistics **2019**)*. Statistics Norway reported a drop of about 10% from 2014-2016, the first-ever drop in direct employment in the Norwegian sector *[9] (Norskpetroleum **2018**)*. The US production in the same time span saw a decrease in crude oil production of 10% *[10] (Patterson **2019**)*, and the Norwegian sector saw an increase of about 1.5%. Relatively speaking compared to employee count, the production has thereby increased in this time span. This show that production per worker, at least in these two countries exemplified, is increasing.

The extraction from unconventional sources means more work per barrel, which again means that each oil engineer has to go cover more ground than before. An example of the increased workload in the industry is discussed by two employees from the Pacific Gas and Electric Company. In their paper, they discuss the consequences of a newly invented simulation technique. This technique involves doing a high volume of batch simulations of hydraulic models and is used in the planning phase of optimizing a pipeline system. They argue that the dramatically increased volume of simulations necessary for this specific method will make a substantial workload increase for the planning engineers. In order for the engineers to keep pace with this increased workload, they suggested the making of an automated tool for the task *[11] (Kolnowski, Ezersky,* et al. ***2016****)*.

For the engineers to be more effective and to be able to meet their delivery expectations, they can not be limited by an outdated computer system. Manually processing the amounts of data today, is highly impracticable *[12] (Bole, Powell, Rousseau,* et al. ***2017****)*. Modern systems have to be adaptive and include system-wide integration across multiple domains, so that as much work as possible can be done and tested automatically *[13] (Dusterhoft, Strobel, Szatny,* et al. ***2012****)*. The introduction of these more advanced digital tools comes with a catch, however. They often demand precise routines and cooperation in order for the system to be functional. Updating the system's data requires dedication into maintenance, and you still have to make sure that the data comes from a trustworthy source. Given this, one can expect automatic testing of the entire system whenever changes are being made. One can expect the generation of dynamic reports for discussions, especially in the planning phase. Live comparison and learning from earlier projects are also possible. And no longer is there a need for creating documentation documents that are both tedious and prone to typos when done manually *[14] (Al-Jasmi, Choudhuri, Joy,* et al. ***2014****)*.

The increased difficulty to both find and extract new oil from unconventional sources, combined with a fluctuating worker count, and a pressure to maintain production, are all contributing to lowering the margins for each barrel of produced crude oil. Even more so is probably the recent years' fall in oil prices *[15] (Mohaddes and Pesaran* ***2017****)*. This is one of the reasons why there is an increased focus on maximizing outcome per worker by utilizing new technology. This in order to be able to maintain the previous production, even with the lower oil prices. An analysis report created by McKinsey & Company describes the future for the oil industry to include the introduction of agile organizational structuring, fluid teaming, looser hierarchies, rapid prototyping, and instant feedback *[16] (Handscomb, Sharabura, and Woxholth* ***2016****)*. Teams will have to adapt to freer structures, with shorter lifespans, and quicker deliveries. The teams also have to have a dynamic way of doing their deliveries, which require having access to tools for both sharing and cooperation *[16] (Handscomb, Sharabura, and Woxholth* ***2016****)*.

2.2 Industry Digitization

This section will describe the current status of digitization across multiple industries and what implications it may yield for the future of the oil industry.

Information Overload

In decision making one often needs to create simplifications of reality in order to be able to form a decision. From birth, we start learning to filter out the information and classify the phenomena we observe. The world has in reality too many variables for us to process. Taking it all in can leave us paralyzed. In truth, most of these variables do not contribute in any way to the making of the decision. This filtering process in real life does, however, come at a cost of a judgment bias in us subconsciously, choosing what data to process and which information to compare *[17] (team* ***2005****)*. The amounts of data produced in the world today are to an extent creating much of the same situation as described above. The difference is that you are comfortable and used to ignoring certain details in real life, but this becomes more difficult when operating on data documents.

Today, the challenge of managing data often lies in being able to utilize the information from the document based data, relevant to the user's query *[12] (Bole, Powell, Rousseau,* et al. ***2017)***, and consequently filter out what can be regarded as noise. *"(...) technological advances today allow more information to be collected more easily and then stored. This information must be exposed in the right way" [12] (Bole, Powell, Rousseau,* et al. ***2017).*** Computer systems have increasingly tackled the information overload problem, which every year grows due to the technology advances that allow higher capacity to store and collect data. Analysis has estimated that the global data amount will double every two years *[18] (Gantz and Reinsel **2012**).* Successfully handling this problem in the same way as humans do, however, only replicates how the decision making happens in real life. You are not utilizing the full potential of the computer age with analysis and processing, and you are still victim to the judgment bias introduced by both the operator and the system's creator.

Finding the correct document to answer the query with the same approach as humans, is a valid approach of using the large amounts of data collected. It does, however, not utilize the full potential of data-rich systems. Other possibilities include the use of new computer systems that allow for heavy processing of vast amounts of data, and one can even use the data in simulating reality. You could by doing simulations have *"(...) the opportunity to mitigate the propagation of incidents into unsafe or uneconomic scenarios when something unforeseen happens" [12] (Bole, Powell, Rousseau,* et al. ***2017).*** In a risk-oriented industry, such as the oil industry is, you could see a great economic and humanitarian gain in the implementation of an all-encompassing system that can handle tremendous amounts of live data. Especially given the oil industry's property of having an inversely proportional relationship when it comes to the pay off ration of planning the well and drilling it. Meaning, investments in planning saves proportionally in multitudes later when drilling begins *[3] (Mitchell, Lake, and Petroleum Engineers **2007**).*

To put the oil & gas data accumulation in perspective - let us say you go out and buy a laptop with the largest storage capabilities commercially available. In Norway, as of May 2019, this results in a machine capable of storing up to 2 terabytes of data. There exist larger disk drives out there, but as of today, this is on the edge of what the marked expect as enough bytes for a private individuals laptop. Seismic scans in the oil & gas industry, for example, produces 6 terabytes of data each day *[19] (Pecholcs, Al-Saad, Al-Sannaa,* et al. ***2012).*** Filling the largest capacity laptop in a matter of hours. A complete scan for an offshore survey in the Gulf of Mexico produces between 1-2 petabytes - 1 petabyte being 1024 terabytes *[20] (Jacobs* et al. ***2018).*** Globally it is estimated that the oil sector alone sits on 1.5 exabytes of seismic data *[20] (Jacobs* et al. ***2018).*** Again multiplying by a factor of 1024 going from one exabyte to a petabyte - making the total amount over 1.5 million terabytes. Buying the largest commercially available hard-drives at the point of writing this thesis, which also gives you the most storage per dollar, would set you off about 58 million American dollars in order to cover the industry's storage need for seismic data[3]. If you were to buy all these hard-drives, you would have a pile of hard-drives reaching close to 3 kilometers tall[4], more than twice the height of the tallest mountain in the United Kingdom[5].

"The Oil & Gas industry has within the last decade formalized this need (information flow for all actors), which has led to the emergence of Information Management roles. However, research suggests that information capture is still challenging to complete successfully even when specified contractually" [12] (Bole, Powell, Rousseau, et al. ***2017****).* This means the ambition is a system that can simulate every aspect of the system, and make use of all available data, in order to give the correct depiction of reality. The next generation of industrial system will certainly aim towards being dynamic, real-time, and all-encompassing. *"This requires effort on all sides to understand what and how we should be addressing this challenge (management of information)" [12] (Bole, Powell, Rousseau,* et al. ***2017****).*

Knowledge Management

Handling large amounts of raw data could lead to information overload. This means the abundance of data makes you either paralyzed in the management for this data, or the processing of the data becomes practically impossible. One should try to capitalize on making the vast amounts of data into something more manageable and thereby valuable. This could be done by creating knowledge from said data. The concept of knowledge is difficult to both define and measure. As Scarborough & Burrell puts it: *"Knowledge is a slippery and elusive concept, and every discipline has its own secret realization of it."* [21].

A reasonable breakdown of knowledge, with a business mindset, is to separate knowledge into explicit and tacit knowledge. Explicit knowledge is actual explicit data. It could be written manuals, product documentation, and handbooks i.e. knowledge that can be read and understood. Whereas tacit knowledge is the knowledge that the employees possess through experience which is difficult to transfer explicitly i.e. internalized knowledge. Nonaka and Takeuchi write in their paper that one should externalize internalized knowledge, and vice versa, thereby entering what they call a *knowledge spiral*. This is one way a company can gain a competitive advantage. Thus, one should employ a knowledge creating mindset in a company by sharing knowledge *[22] (Nonaka and Takeuchi* ***1998****).*

The increased workload in modern businesses calls for an expectancy to process more data when creating new explicit knowledge. This is difficult for those who lack the necessary tacit experience and training *[23] (Smith* ***2001****).* The correct handling of data requires expertise of the system and how to act upon the data in a decision-making process. This expertise i.e. the experiences and know-how, in combination with this data, creates knowledge. Knowledge is thereby closer to the actions that will be done than the sum of its parts. It is used when making decisions in a company, which makes it a crucial corporate asset *[24] (Davenport, Prusak,* et al. ***1998****)*

[3] Toshiba 14TB hard drive for 517.5 USD

[4] The drives are 26.1 mm tall

[5] Ben Nevis - 1 343 m

> *Knowledge is neither data nor information, though it is related to both, and the differences between these terms are often a matter of degree.*
>
> - Thomas Davenport & Lawrence Prusak
>
> *[24] (Davenport, Prusak,* et al. ***1998****)*

As a company grows, so does the data and information they possess. To utilize the growing amount of data and information, one should employ mechanisms to make sure that its potential to increase the company's knowledge does not go to waste. Thereby, a company has the need to *orchestrate* data and information into something that could be used *[25] (Monteiro, Pollock, Hanseth,* et al. ***2013****)*. As data is stored in different forms and manners, it helps to structure these consistently. E.g. physical papers are more easily read if they have visual cues which help their readability *[26] (Nygren and Henriksson* ***1992****)*.

Complexity Management

A system for managing the planning of a new oil well, fall under the category of complexity management and Product Life-cycle Management (PLM). A complex system is often defined as a system that has a lot of connected components that is difficult to comprehend and predict *[27] (Cotsaftis* ***2007****)*. Complexity management often includes multiple fields of expertise working together towards an as good as possible solution. Product Life-cycle Management is a business model that aims to better the collaboration in product design, product manufacturing, and help with cooperation with the different business actors *[28] (Tao, Wang, Zuo,* et al. ***2016****)*. The demand for having these tools created, in order to stay competitive, have pushed the use of technology forward and rendered the use of these tools an absolute necessity in many businesses. Digital tools can often have revolutionary or even disruptive effects when they solve problems in the order of multitude in terms of efficiency. Just imagine having to do manual accountant work for a large cooperation today. Some tasks like manual accounting are today not practical and in some cases not even possible, without the proper set of tools. Doing the same job without the help of software and computers would have required a whole staff of employees.

> *The old approach—using disconnected, legacy PLM systems, outmoded tools like spreadsheets, and one-of-everything IT architectures, simply won't work when it comes to making products to meet tomorrow's demands.*
>
> *[29] (Aras* ***2018****)*

Tools that enable the increased delivery from each worker, and that can enable the complexity to be easily communicated, is increasing in demand also in the oil industry. Especially with the increase of IoT devices constantly producing data. Many of these systems are in the early stages of development and are not entirely deployed. The continuation of this section will, therefore, look to other industries in order to get a better understanding of the possibilities and the errors that already exist in other complex management systems.

> *There is no need to access multiple systems to find engineering information. (...) You don't have to access three or four systems to find the engineering information you need for the job. (...) It's all here, which is a big advantage*
>
> - Geir Sjøsåsen
>
> Using digital twins in the oil & gas industry
>
> *[30] (Sjøsåsen **2017**)*

Industry of Architecture, Engineering, and Construction

The Architecture, Engineering, and Construction (AEC) industry got introduced to the concept of Building Information Modeling (BIM) in 2002. The concept builds on ideas originating from the 80s, but it was only now in the early 2000s that software started to implement some of the ideas. BIM's exact definition has been redefined several times since its original introduction. It started as a tool with a focus on integrating design and modeling with engineering. A simple tool for visualization in 3D with exact data and measurements. Typically with the use of AutoCAD software. This is in the UK known as BIM level 0 and level 1. It has since evolved to be an umbrella term for an object-oriented, three-dimensional modeling tool, that facilitates interoperability and information exchange through collaborative features *[31] (Miettinen, Kerosuo, Korpela,* et al. ***2012****)*. It is a unifying repository of project information and graphics that aim to reduce design mistakes and increase the productivity of a project *[32] (Miettinen and Paavola **2014**)*. This is now typically what is known as BIM level 2, as it includes all players involved in a project, not only the explicit contractor. BIM is today a well-known term in the industry, and it is also currently one of the most promising innovation to tackle the industry's performance issues *[33] (Cao, Li, Wang,* et al. ***2017****)* with regards to the rising complexity of diverse projects *[34] (Ghaffarianhoseini, Tookey, Ghaffarianhoseini,* et al. ***2017****)*.

One of the goals of modern BIM today is that it encompasses all necessary software features into one coherent system *[32] (Miettinen and Paavola **2014**)*. The technology promises components that are shareable in a collaborative manner, to improve all processes throughout a project's life-cycle *[34] (Ghaffarianhoseini, Tookey, Ghaffarianhoseini,* et al. ***2017****)*. BIM's ability to transform an organization's collaborative mindset has however been slow to emerge *[32] (Miettinen and Paavola **2014**)*. The technology does promise increased collaboration and knowledge sharing across both interdisciplinary and organizational boundaries. But conservatism towards using unknown software in a risk-averse corporate culture, combined with the general reluctance to change, may be the root of the slow adaptation *[35] (BCG **2018**)*. There is typically an employee resistance in going from the known to the unknown *[36] (Bovey and Hede **2001**)*. Unknown in this context could mean a fear for the future of the individuals' position, due to automation, but it could also be a fear of not being able to successfully execute the new work task *[37] (Robey and Sahay **1996**)*. In general, individuals seek to maintain a state that is stable in terms of stimulation *[36] (Bovey and Hede **2001**)*, and would otherwise require the proper motivation and exposure in order to successfully adopt a new system *[37] (Robey and Sahay **1996**)*.

Problems with employees' conservatism and risk-averse corporate culture could be the reason for the slow adoption of new technology, but there is also evidence that the AEC industry as a whole has their own set of specific problems. It has, in the UK, been reported as highly fragmented, with limited communication and a lack of trust *[38] (Egan **1998**)*. This affects the sharing of knowledge and experiences *[39] (Chinyio and Olomolaiye **2009**)*. A lack of trust by the industry's actors may limit the adaptation of the wanted collaborative tools and hence also the adoption of modern BIM. It is also a valid point in that the potential of BIM, and what it promises, is not well communicated or measured *[35] (BCG **2018**)*. When working towards change with multiple actors across the whole value chain rather than only a single corporation, it can be difficult to establish new industry standards. New *de facto standards*[6] often come from innovative solutions that are more successful than others. But with a technology that demands cooperation in order to succeed, you will struggle to get the ball rolling in the first place. You have to have other incentives that force a collective change. That is why the industry's largest clients, namely governments, have to take the lead in promoting the use of BIM *[35] (BCG **2018**)*.

The UK government decided a mandate in 2011, that all government projects from 2016 and forth should use at least BIM level 2. It is an effort to increase the use of BIM with the aim of reducing cost, reducing risk, improve carbon performance, and improve planning estimates. It is also supposed to reduce the total cost of public sectors assets by up to 20% *[41] (BIS **2019**)*. The hypothesis for this decision includes: increase value for no extra cost, establish non-proprietary formats following international standards, increase competition, and the introduction of transparent documentation *[42] (BIS **2019**)*. These changes are beneficial for both the industry and the consumers, but would maybe not been plausible had it not been for a coordinated effort through this mandate. Figures show that the industry in the UK as a whole went from 13% usage of BIM, with an awareness of 58% in 2016, to in 2018 having a 74% usage and a 99% awareness. At least 58% claim they have reached level 2 or higher of BIM *[43] (NBS **2018**)*.

Industry of Machine Engineering

The fourth industrial revolution conceptualizes the next step of automation and use of data in manufacturing. An approach to realize some of this potential is the all-encompassing technique of *digital twins*. El Saddik defines digital twins as *"(...) a digital replica of a living or non-living physical entity. By bridging the physical and the virtual world, data is transmitted seamlessly allowing the virtual entity to exist simultaneously with the physical entity" [44] (El Saddik **2018**)*. It started much like BIM through the idea of representing objects of interest as digital CAD models[7]. The difference is, however, that a digital twin aims to be an accurate virtual representation of both the elements and the dynamics of

[6] *"A set of criteria for software, hardware, or communications procedures that are widely accepted because of the dominance of a particular technology over others rather than the action of a recognized standards organization" [40] (Parker **2003**)*.

[7] CAD=Computer-Aided Design

an object. It is modeled to behave realistically to all simulated situations. Following the UK standards, it is a step towards the undefined BIM level 4 covering so-called Enterprise BIM. A digital twin requires you to fill in all properties of the object before production can begin. Later when the object is completed you continue gathering all valuable information regarding the object properties in real-time and feed it back into the model for calibration. This loop continues and makes changes to production and use throughout the whole life-cycle of the object. The data collected can later be reused and improve later projects *[45] (Post, Groen, and Klaseboer **2017**)*.

Aaron Frankel, Director of Product Marketing Manufacturing Engineering Software at Siemens, talks about the future of the production industry in a product demonstration in 2015 *[46] (Frankel **2015**)*. He states that tolerances in manufacturing are getting more critical as the enclosures for the items produced are getting smaller and more precise. The difficult operation does not lie in creating a good design of a product, but rather the almost impossible task of coming up with a great way of producing it. This is almost the exact words that Tesla Inc. CEO Elon Musk used to describes their difficult situation when producing the Tesla Model 3 in 2019 *[47] (Rapier **2019**)*. The advancements in technology make it possible to simulate hundreds of productions with error margins to optimize the product for actual production. These simulations can be done in minutes with hundreds of parameters and alterations. This saves money as the process is much faster than the traditional method of physical trial and error, and creates a great competitive advantage in getting the product quicker to market.

When having a complete, all-encompassing system that digital twin promises, you can also run simulations in how the production affects the environments in which it is being produced. You could, for example, look at the production with the use of a workforce of either humans or robots, and generate a risk report of health and mechanical wear *[46] (Frankel **2015**)*. Small tweaks to table height or grabbing length in the production area could then be simulated and analyzed in order to improve the work conditions. The system also enables the automatic generation of diverse manuals for the individual workers doing their specific task. This eliminates the tedious task of keeping documentation up to date when doing minor modifications to the production system. In fact, all parts of the system promise to be up to date with the latest modifications implemented, as all data is interlinked *[12] (Bole, Powell, Rousseau,* et al. ***2017**)*.

Another advantage of having a complete system of digital twin, is that it brings all of the relevant actors together, not unlike what is seen in BIM and the theory behind PLM. Due to the single, all-encompassing system, you will have an easier time accessing all of this information as it should be available to all actors for updates. Consequently, this ensures a greater cooperation between the separated actors as they are forced to work and enact on the same set of data. The system should also enable the information to flow much easier if system parts are being updated and your responsible component is affected. Aaron Frankel exemplifies by explaining that digital twin brings designer and manufactures together, which have historically been known to have a limited communication *[46] (Frankel **2015**)*. In order for a system that is implementing digital twin to succeed, you have to have a cooperation with these two actors, and all the actors of the system, throughout the product's life-cycle. They have to feed the system with all its required data for the mutual benefits of increased innovation *[48] (Van Os **2018**)*. Without this, you will not have a system that responds realistically in simulations and estimations. There is, therefore, a great deal of network externality factors to a system of this sort.

Network Effect

A major challenge in implementing some of these new systems that exist for cooperation lie in getting a substantial user-base. Especially systems like digital twin, which is never stronger than its weakest link in the chain of actors. Meaning, systems that do not get their value without its community. This is a situation described as network effect and a problem which can be expressed as a bootstrap problem. In the bootstrap problem, the creators of the system have to somehow persuade users into adopting a system that does not serve its purpose without more users. An effort is required from the system creators to create other benefits that can persuade the user into adopting the system. Benefits that function regardless of the number of users *[49] (Hanseth and Lyytinen **2010**)*. An example of this problem faced the online community of Reddit Inc. In Reddit's early days of 2005, its founders were forced to create false profiles and manually publish what they regarded as interesting content on the page. This was as a mean of getting the community started. Unique for Reddit, compared to the systems implementing digital twin, for instance, is that they only needed a handful of users before the system would be self-sustainable and growing on its own. This is what is known as reaching competitive sustainability, implying that success exponentially begets more success *[50] (Tiwana **2013**)*.

2.3 Modern Software Development

Web Technology

The introduction of supportive web development frameworks like Angular, Vue, and React in the 2010s have made it easier to develop large web platforms that are dynamic, effective, and available for hosting in the cloud *[51] (Roby, Wu, Goldina,* et al. ***2016**)*. These frameworks allow the application to be single-page, meaning only elements that are different will change when navigating through the page. This gives the application faster response times and limits data consumption *[52] (Shahzad **2017**)*. Tools for managing complex systems, as described earlier in this section, can benefit from this technology in making the applications available online *[53] (Schroeder, Steinmetz, Pereira,* et al. ***2016**)*. A well-written application can replace the need for traditional applications that are installed on individual computers. The development of automatically scaleable solutions also offers dynamic computational power if needed. The move towards cloud-based computing means there is little to no benefit in having these locally installed programs anymore. The disadvantages are close to none, but the advantages are many. For one, you have the possibility to start your application directly from any machine hooked up to the internet *[54] (Alam and El Saddik **2017**)*. Possibly with an application that is tailored for screens of mobile devices and touch, you could be able to access any program on the go *[53] (Schroeder, Steinmetz,* *Pereira,* et al. ***2016).*** This makes the sharing of information easier between businesses and enables digital cooperative tools inspired by modern development approaches.

Agile

Many companies and practices have changed ever since the introduction of Agile in the world of software development in the early 2000s. Agile launched with the agile manifesto, containing 12 simple principles to follow[8]. These have later been an inspiration to many parts of corporations, not only software development. The term *"agile"* can often be seen in conjunction with management, manufacturing, construction, and enterprise *[55] (Chen, Reichard, and Beliveau* ***2007).*** It has a view on handling development risks through continuously improving the product in small increments, rather than intensive planning. This ensures that, at any point in time during mid to late development, you have a working product that can be delivered, often called a *Minimum Viable Product (MVP).* It helps with presenting the progress to the different stakeholders, as you have a working demo to show, increasingly getting better, rather than bits and pieces of a product. This tends to help excite project stakeholders and keep developers focused *[56] (Ferreira and Cohen* ***2008).*** It also ensures that if something were to happen to either time, cost, or scope, you will be able to still deliver a working product.

Agile embraces change by pressing for continuous delivery. This is in high contrast to the traditional method of waterfall development where you begin the project by tedious planning and the creation of a development schedule that is to finish on a set deadline *[57] (Sommerville* ***2011).*** Agile also emphasize that the individual team should be self-organized. Meaning; teams should be able to choose what project to take on and how to execute it, without having to wait for a manager to assign work or set deadlines. This helps the team being more responsible and gives them a higher sense of ownership.

With Agile came a series of supporting frameworks to help teams in organizing for maximal efficiency and guide them in the following of the agile manifesto. The most prominent of these frameworks include Scrum, Extreme Programming (XP), and KanBan *[58] (Matharu, Mishra, Singh,* et al. ***2015).***

- **Scrum** have distinct role allocation and a set structure of meetings in order to help the communication and development process. Directly responding to the agile manifesto's sixth and twelfth principle: *"6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation"*, *"12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly".* Scrum also introduces the notion of working in sprints, where each sprint will deliver a working version of the developed software. This helps in fulfilling agile's promise of continuous deployment and delivery with high frequency.
- **Extreme Programming** consists of twelve practices including pair programming, collective ownership, continuous integration, on-site-customers, and cod-

[8] `agilemanifesto.org/principles.html`

ing standards. These practices have shown to be very effective *[57] (Sommerville **2011**)*, *[58] (Matharu, Mishra, Singh,* et al. ***2015****)* in defining and concretize self-organization, in keeping team members motivated, and it covers agile's principle mentioning the inclusion of business people.

- **KanBan** introduces a framework for managing tasks in a prioritized backlog. It organizes the tasks into categories of completion. It helps the team in having an improved overview of the tasks at hand, enhances mutual understanding of each team member's contribution, and it enhances the overall workflow[9].

Agile Implementation In Practice

The employee recognition company O.C. Tanner made the leap to being agile in early 2002. O.C. Tanner produces over 9000 recognition awards daily and delivers to customers from 150 countries. The IT division of O.C. Tanner only made up a total of 10% of the entire staff at the time, but it was nevertheless decided to implement the principles of agile across all parts of the company. One of the problems with the company was the lack of synchronization between IT, responsible for sales and agreements, and the actual production. A step in fixing this was done by letting both divisions use the same software system to coordinate and plan. They implemented cross-team communication channels, and the production division started having regular stand-up meetings *[59] (Cockburn **2006**)*, *[60] (Atlassian **2006**)*, *[61] (Gostick **2004**)*.

Implementing the principle of having the employees be self-organized meant the management could take a step back and interfere only when absolutely necessary. Employees can in most cases collectively make the best decision, and if not they will gain experience and learn from their mistakes. Managers should thereby act more as leaders than to directly manage and dictate the employees. One of the manager's new continuous assessment is, therefore, to let employees do mistakes for the purpose of learning, and at the same time look after the company's well being. There will be situations where the manager's extended experience makes them able to see potential outcomes that are unknown to the employees. This little change of responsibility made a considerable change to the entire company structure. By making the teams be self-organized, you have also changed the role of management to follow the Agile principles.

Part of why O.C. Tanner succeeded in their change to the agile principles were that the introduction of changes happened gradually. Instead of demanding certain impulsive disruptive changes to the whole company at once, they tested out the change in smaller parts of the company. Even though management could be absolutely certain that this change would excel the company moving forward, they still let the employees get accustomed to the strategy and experience its benefits. Just like the pilot initiative proposed by Orlikowski in his paper on organizational issues *[62] (Orlikowski **1995**)*. This allowed the employees to directly compare the strategy's performance through diverse metrics of measuring success. This can be looked upon as an incremental improvement of the company structure itself. An initiative that boosts the collective thinking, and thereby also the company culture.

[9] `https://resources.collab.net/agile-101/what-is-kanban`

In terms, this aided the teams in staying self-organized, while still enabling strategical changes to be executed from the upper management. Making the management lead the way forward, rather than directly managing it by dictation. The need for techniques like this really comes down to our built-in resistance for change and our risk-averse corporate culture whose ideas build upon the same principles. By showing environmental ideas in smaller increments in practice, and by inspiring towards a specific direction of strategy, you hinder the disruptive nature of episodic changes. It supports the fifth principle of agile: *"Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done"*.

When doing changes, and particularly changes to a system, one has to be aware of the users' motivations across the organization. There is often not the change itself which is the problem when issues arise, but rather how the process and introduction of said change are executed. An example where the implemented system is the same, and the organizations are more than comparable is described by Robey and Sahay *[37] (Robey and Sahay* ***1996****)*. Their case looks at the implementation of a particular geographic information system in two neighboring government counties. They explicitly showed in their case how important a combination of inner motivation, proper documentation, employee training, system knowledge, and the continuous transition was in order to succeed in an organizational change. Similar in many aspects to the success of implementing agile in O.C. Tanner.

3 Case Description

This section will describe the case of this master's thesis, namely the feature Well Montage. It introduces the necessary context for the case, a description of the platform DrillPlan, how its feature Well Montage operates today, and how the feature is lacking in functionality.

3.1 Case Context

Well planning is, as discussed in the previous section, a tedious and complex operation with large amounts of data that needs to be processed in order to reach the goal of a *safe*, *usable* and *minimum cost* well. A drilling program is a long and detailed document, which contains, if not all, but most of the important information regarding the well. Depending on the project and the well that is to be drilled, it could be up to hundreds of pages long. Table 1 illustrates the differences in the types of wells that are made and the size of the drilling program. *Small wells* are low-cost and highly repeatable wells. They do not require large amounts of planning, and its main operation can function with as little as a single page of specifications. *Medium wells* are increased in size and complexity and, consequently, expects higher profits and lifespans compared to the smaller wells. They will require more of the exploratory nature, described in Section 2.1, and thus needs a longer time to plan out. *Large wells*, often offshore wells, are huge projects both in scale, employees and complexity. They can take years to plan out, and produce millions of pages of documentation *[63] (Benjamin **2018**)*.

Size of the well	Length of planning	Pages of drilling program
Small	Days	1
Medium	Months	75
Large	Years	200

Table 1: Complexity of well planning based on size. Data collected from interviews and *[64] (Valusek **1999**)*, *[65] (SLB **2018**)*, *[66] (Landmark **2012**)*

3.2 DrillPlan

To alleviate the complexity that is well planning, Schlumberger introduced DrillPlan. DrillPlan is the first web-based well planning software that aims to have full coverage of the well planning process. It focuses on enhancing the well planning process by allowing for collaborations between the actors, of the company that is using DrillPlan. All your data and calculations are done through this software and are stored and available from the cloud. It reduces the former need for specially dedicated and powerful desktop computers by having distributed computational power available for all users of DrillPlan as long as they have an internet connection. It is unique in the way that it is the only all-in-one tool on the market. Meaning, all the steps from well planning, as seen in Figure 1 on page 4, is packed into one coherent package, and the data intercorrelates with each other making the well planning more efficient. By having an iterative work process of the well planning, it increases the efficiency by fully utilizing a cloud-based and collaborative work platform. In addition to this, it automates repetitive tasks and validates the inputted data. By having all of these features in one coherent package, it makes DrillPlan a unique complexity management tool for well planners *[67] (Schlumberger **2018**)*. It also supports the second digital age that the oil industry is currently going through, by aiding the industry with additional assistance from smarter and more efficient digital tools *[68] (Choudhry, Mohammad, Tan,* et al. ***2016****)*. An example of this is the *automatic engineering analysis*, as illustrated in Figure 2. It runs continuously and automates calculations of the validity of the data that is inputted for the chosen well.

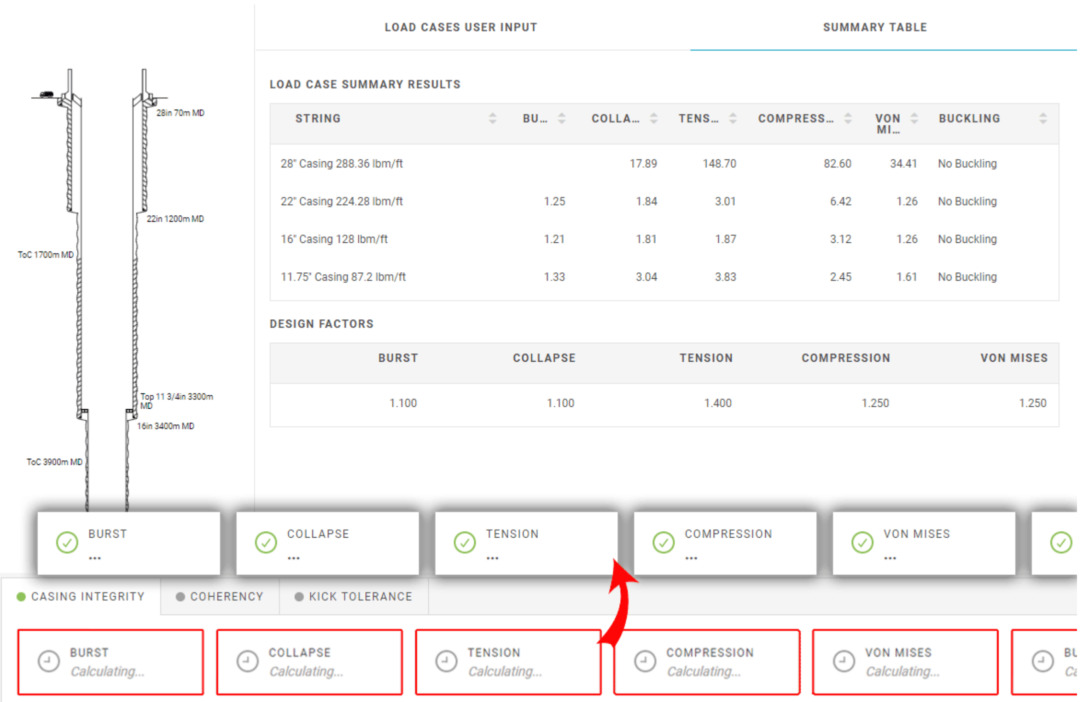


Figure 2: Automatic analysis running (marked in red). Visuals shown after calculation completion is marked by the arrow. On the left side of the figure is a automatically generated graphics showing casing details from the data tables.

The Well Montage

The entirety of the useful data in a DrillPlan project can be boiled down into a drilling program. To combat the triteness of having this large document with excessive amounts of data and information, the industry often creates what is known as a *one-page summary.* A one-page summary is a compilation of the already compiled drilling program onto a single page, often in size A3 or A2. It includes the most essential fields such as the design of the casing, BHA, and mud properties. There is no definitive answer as what to include or how a one-page summary should look. The variety of how one-page summaries can be graphically different, is illustrated in Figure 4.

On request from the users of DrillPlan, Well Montage was introduced as a feature to DrillPlan in the summer of 2018. It generates a one-page summary, from data inside of DrillPlan, and exports this to an excel document that fits within a single printable page. An example generated from within DrillPlan can be seen in Figure 3. The implementation of the Well Montage feature was not planned to be developed at this point, but it was added due to heavy user feedback, advocating the necessity of its functionality. The lack of established standards on the matter meant that Schlumberger had to map out these themselves. They did this by comparing four one-page summaries from different companies. The comparison identified 12 different categories of information. Within each of these categories was an even larger list of attributes, also called data fields. Which of these attributes to include in DrillPlan's Well Montage were then picked out based on a refined list sorted by their frequency of use. The number of attributes to be included were restricted by the free space available on a single sheet of paper.

The establishment of what data fields to include in Schlumberger's Well Montage included 54 unique attributes spread to 12 different categories. This is a highly compromised list of attributes given the total of over 140 unique attributes identified in only four compared documents. The omitted attributes obviously have their function, or else they would not have been used. In addition, each document that was compared had a different layout and structure which was not accounted for. Today's solution is a product of the elements that is the closest fitting to the four one-page summaries given. Any customization needed - may it be data modifications, tweaking of visuals, or even comments - have to be done offline in a spreadsheet editor like Microsoft Excel, completely outside of the dynamic platform of DrillPlan.

The current state of DrillPlan's Well Montage is therefore too rigid, and users are requesting more customizability. This includes additional attributes to be added, and possibly specific layouts without having to manually update. It is therefore in Schlumberger's interest to make a more modular and configurable Well Montage to satisfy the users' requests and to better fit their dynamic DrillPlan web-platform. The complete master's thesis proposal can be read in its entirety in Appendix A.

Contribution

Little research on the concepts and explicit information on the use of the one-page summary exists. To our knowledge, only one paper describes in details the explicit content of a one-page summary *[69] (Sadhwani, Sanderson, MacFarlane,* et al. ***2016****)*. Other papers write about the benefits of using similar one-page summaries, with regards to increased project performance and the knowledge creating aspects of them *[70] (Rowe, Keasberry, Hunt,* et al. ***2017****)*, *[71] (Al-Suwaidi, Faure, Talal,* et al. ***2002****)*, *[72] (Menhali, Mohammed, Sulaiman,* et al. ***2013****)*. However, they describe the use of the one-page summary as one of many benefiting factors for their increased performance and does not go into details as to its design.

This master's thesis will try to shed light on an often used, but non-standardized, part of well planning. We will try to map what a user is looking for in a new and improved Well Montage and make a prototype to fit their needs. In addition, with the aid of DrillPlan as an interactive web-platform, try to implement the next version of Well Montage that takes the users' needs into account, while utilizing the power of an interactive web experience. Thus, contributing with a proposition of how and why the use of a one-page summary is a necessary part of well planning.

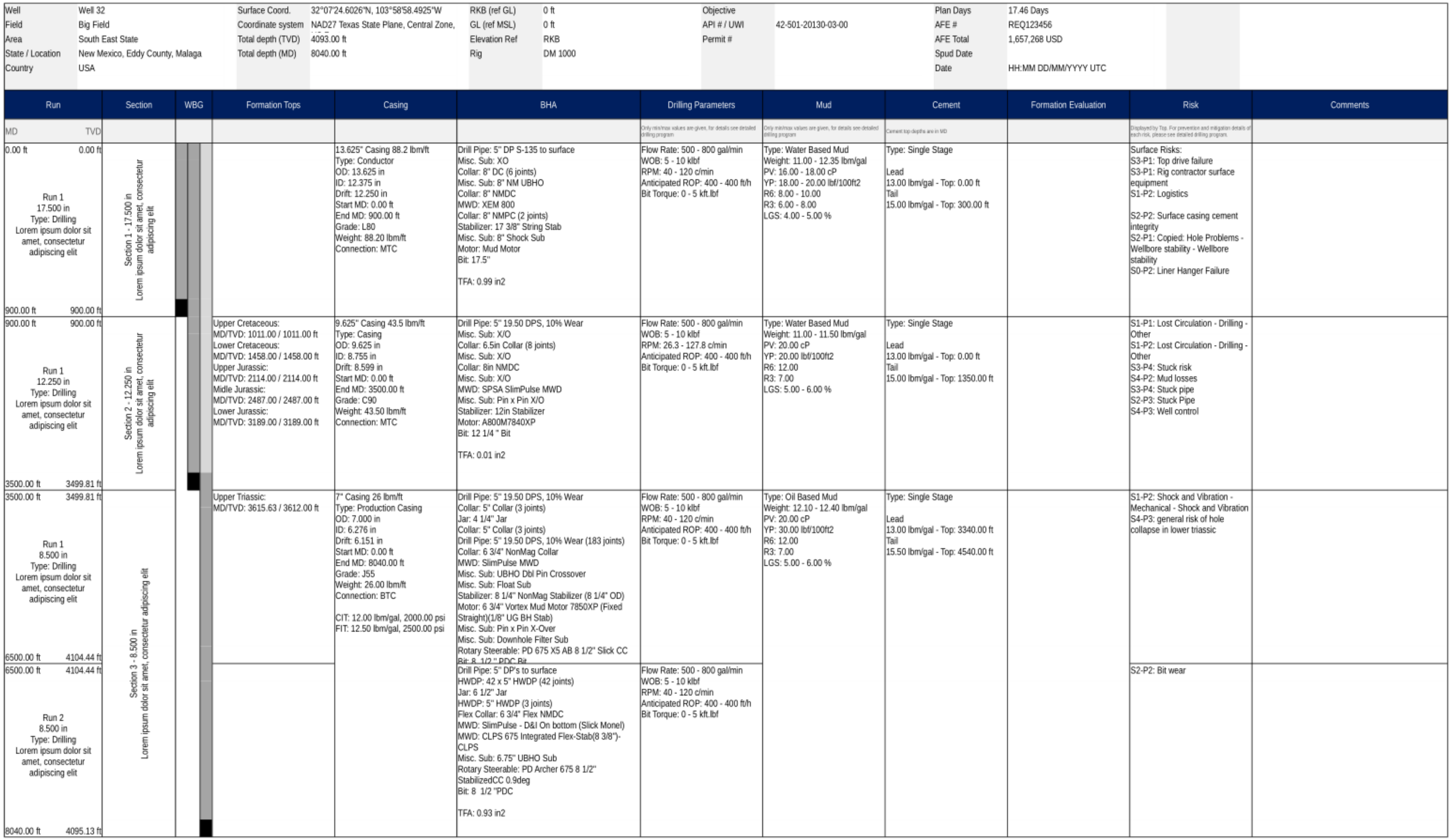


Figure 3: How DrillPlan's Well Montage looked as of September 2018

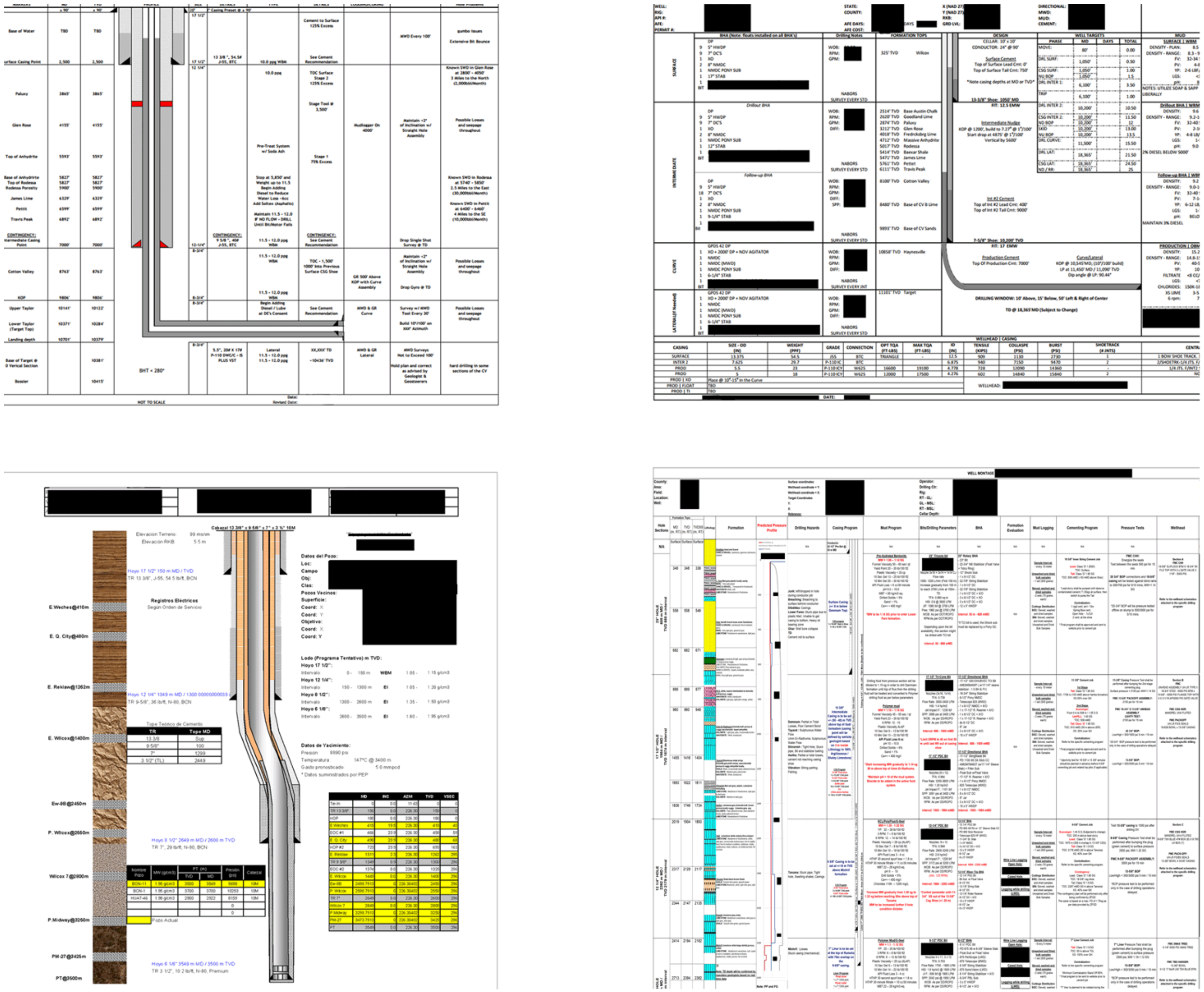


Figure 4: Examples of other one-page summaries used as inspiration for the current implementation

4 Method

This section will present the methods used in this project. It includes methods for requirements and goals engineering, testing, and prototyping. The section ends with a presentation of the field of interaction design, that will be used throughout the prototypes.

Project Scope

Our project has an established user need and a limited team of developers. The *project management triangle [73] (Atkinson **1999**)*, describes a project's quality by the project's cost, time, and scope. Using the project management triangle - this project has a set problem, with a fixed deadline, and consequentially; a dynamic and definable scope. The scope, therefore, has to be set to match the limiting two fixed parameters as best as possible. Meaning, there has to be a prioritization of what to focus on in order to have a desirable end result within budget. What defines a desirable end result is defined by the project goals.

4.1 Project Goals

This project has two main deliveries that require attention. The first goal is the master's thesis delivery. It is expected to be a comprehensive report, containing and describing all parts of the process from start to finish, as well as a rationale of where this project lies in relation to previous research and challenges. The second goal is the final presentation of the developed system for the employees of Schlumberger. Here we are expected to present the whole project life cycle - from idea, to prototypes, to a functioning demonstration. The demonstration will have a more technical focus than what is described in the thesis. Due to the holiday season, the presentation is expected to take place no later than early June, i.e. before the thesis is delivered.

With these limitations, an execution plan could be established. Statistics Norway[10] say in their handbook for *Practical User Testing* that product development, i.e. the road from an idea to a product, comes in two phases. They divide the process into *specialization* and *operationalization*. Specialization of the initial idea is about gathering and prioritizing user needs. The operationalization phase is where you create and test a functioning product that satisfies the user's needs *[74] (Sundvoll* et al. ***2006**)*.

This project was divided into two main phases in accordance with Statistics Norway's handbook.

1. The first phase, the *specialization*, was done in the autumn of 2018. It was an exploratory case study with the goal of researching the concept behind well planning, one-page summary in general, and DrillPlan's own Well Montage. This was done through qualitative interviews with representatives from customer relations, and con-

[10]Norwegian Statistics Bureau (SSB) Statistics Norway has overall responsibility for providing statistics on Norwegian society.

versations with Schlumberger employees. In the end, a proposed set of initiatives was suggested based on the information gathered. All collected information and proposed initiatives were documented in a written report at the end of 2018 *[75] (Einarson and Fredriksen **2018**)*. A further description of the case study can be seen in Section 5.

2. The second phase, the *operationalization*, was done in the spring of 2019. On par with modern development approaches, this phase was done in sprints. We opted for doing three prototyping sprints, as to give time for the writing of the thesis. The first sprint set project goals and scope, and the first and lowest fidelity prototype was tested and evaluated. The second sprint took the evaluations of the first sprint in mind, and an upgraded version was made; both in terms of fidelity, but also iron out design faults from the first sprint. Lastly, in the third sprint, the functional prototype was made to test the most crucial aspects of the program, with code. It would be, combined with the thesis itself, the main delivery to Schlumberger.

4.2 Conceptualization Techniques

Many of the techniques used throughout this paper are highly inspired by the book, *Sprint: How to Solve Big Problems and Test New Ideas in Just Five Days [76] (Knapp, Zeratsky, and Kowitz **2016**)*. The techniques described in this book has been used in Google Ventures and has contributed to the development of e.g. Dropbox, Facebook, New York Times, Google Gmail, and Google Chrome. It settles around having a complete development team consisting of decisions makers, as well as domain expert, to both test and explore a specific problem in just a week. The book guides you from having just a vague idea the first day of the week, to having a tested prototype on Friday evening. The goal of the week, or *"Sprint"* as the book calls it, is to gain more domain knowledge of the problem and to try out new concepts and effectively test an idea, without wasting time and money. It also functioned as a great way to develop relationships within the team, which enables the transfer of knowledge.

Our project, however, differs a bit from most of the cases described in Knapp's Sprint book. The problem we were to solve was not disputed in any way, and no one questioned its demand or necessity to be developed. Unlike what is often a premise for using the Sprint book's techniques. There was, however, a great deal of creative freedom as to both design and technical decisions to our project. That was why we took inspiration from some of the techniques in Knapp's book - namely for boosting the creativity, not for managing the project as a whole. These techniques have been proven effective in mapping out requirements and concretizing these requirements into testable elements in a prototype. The book helped with how to go from a problem or idea to a prototype. It did so by providing specific actions and tasks to be done, in order to progress from the idea phase to something more malleable. In many ways, it was used as a *"encyclopedia"* for when the project needed an extra push, which came in the forms of tasks from the book.

4.3 Methods for Prototyping

The project was planned as three iterations, thereby requiring three prototypes. The prototypes varied in fidelity and completeness, thus, trying to achieve high coverage of all the necessary parts of the system. The project lasted for 21 weeks, and a sprint usually lasts for a maximum of 6 weeks. Since we are just two people, and the additional time needed for documentation, we went for a maximum 6 weeks sprint. Thus having a minimum of 3 weeks to spare. Each prototype would, therefore, get a maximum of 6 weeks to be planned, made, tested, documented, and evaluated.

A case study looking at usability problem identification, specifically comparing low-fidelity and high-fidelity prototypes, showed that low-fidelity prototypes are as effective at detecting usability problems as high-fidelity prototypes. Meaning, the number of identified usability problems were comparable for both approaches *[77] (Virzi, Sokolov, and Karis **1996**)*. The research was not conclusive on the question of whether low-fidelity prototypes find different problems than high-fidelity. The problems identified were mostly the same for both the two conducted experiments in the research. There were a few problems uniquely identified for each of the approaches. The researchers' closing belief, however, was that the two approaches were comparatively effective in identifying problems and that they could be regarded as interchangeable. Suggesting that one should use low-fidelity prototyping where applicable, as it is cheaper than creating high-fidelity prototypes.

Type	Advantages	Disadvantages
Low-fidelity prototype	Lower development cost, Evaluates multiple design concepts, Useful communication device, Addresses screen layout issues, Useful for identifying market requirements, Proof of concept	Limited error checking, Poor detailed specification to code to, Facilitator-driven, Limited utility after requirements established, Limited usefulness for usability tests, Navigational and flow limitations
High-fidelity prototype	Complete functionality, Fully interactive, User-driven, Clearly defines navigational scheme, Use for exploration and test, Look and feel of final product, Serves as a living specification, Marketing and sales tool	More resource-intensive to develop, Time-consuming to create, Inefficient for proof-of-concept designs, Not effective for requirements gathering

Table 2: Low fidelity vs high fidelity prototypes, from *[78] (Preece, Rogers, and Sharp **2015**)*

Table 2 shows the differences between high fidelity and low fidelity prototypes. It describes in what parts of the process they are the most effective. Such as, a high fidelity prototype should not be made for testing an initial idea that has a great chance of being discarded. This would require an unnecessary amount of work, which could have easily been avoided by creating prototypes more advanced in line as the project progresses. Sauer and Sonderegger suggest both an increase in usability and that the test-users' emotions were positively affected by a higher fidelity prototype *[79] (Sauer and Sonderegger **2009**)*. From Table 2 and other research done on the matter of prototype fidelity *[80] (Walker, Takayama, and Landay **2002**)*, one could advocate the usage of prototypes with an increasing fidelity per iteration, as the functionality gets more advanced, also for our project. Thus, complying with the advantages and disadvantages as seen in Table 2. Starting with the lowest fidelity being a paper prototype, and moving to a non-functioning prototype, and finally, a high fidelity functional prototype set in its actual environment.

The first prototype we opted for was a paper prototype. It is a low-fidelity prototype with limited functionality and complexity *[81] (Sarma and Van Der Hoek **2006**)*. As seen in Table 2 low-fidelity prototypes also have disadvantages, so it is important to stray from those by making a paper prototype for its advantages. It is a fast and low-cost way of checking the flow of an application, labels on buttons, or if concepts communicate effectively. It confirms that the application corresponds on a top-level to the users mental model. A particular design can, therefore, on an early stage, be compared and later discarded *[82] (Rubin and Chisnell **2008**)*. This saves money in unnecessary developed code and helps to focus the vision early on.

The benefit of having a crude paper prototype, apart from saving time and effort under creation, is the enhanced feedback it may yield. Testers have a tendency to give less feedback when the product has a well-developed interface, as this affects their emotions positively *[79] (Sauer and Sonderegger **2009**)*. It is harder to critique a seemingly finished product than a low-fidelity interface with obvious shortcomings. The error tolerance of the feedback is much lower when discussing an unfinished paper prototype, making it less daunting to contribute, and you can, therefore, request and expect more direct assessments from the users *[83] (Vijayan and Raju **2011**)*.

Our creation of the paper prototype began where the requirements engineering ended. With the prioritized list of features, we could then have a grasp of what to develop. When creating a paper prototype one must understand to not design a complete and finished user interface. It is supposed to be quick and discardable. One should rather focus on core elements - high risk and top-level features that need testing. The process of deciding what to test in our project came about by following the first steps of Knapp's Sprint book, namely to create maps describing core functionality.

Exercises for creating drafts of layouts can be carried out when an adequate map describing core functionality to focus on is created. We followed another exercise described in Knapp's Sprint book for effectively producing new ideas and for the idea process to be efficient in terms of time investment. The exercise simply asks the team members to draw very fast and crude drafts of user interfaces on paper. The goal is to create as many different solutions as possible over a limited time period. It is normal to divide the exercise into smaller activities focusing on specific sub-parts of the system.

In an effort to orient what interactions other user interfaces have, we created a document with inspirations, which can be seen in Appendix E. It contains useful advanced user interactions and ideas possible to add in our prototypes. As Pablo Picasso puts it: *"good artists copy, great artists steal"*. Meaning, why try to make something entirely new, when you can use something that users are already familiar with *[84] (Kleon **2012**)*?

With regards to how to do prototypes with higher fidelity, there are not as many clear guidelines. As seen in Table 2, high-fidelity prototypes should be fully functional and serve as a *"living specification"*. Since the project was split into three iterations, we opted for a medium-fidelity prototype for the second iteration. Trying to encompass advantages and disadvantages from both low- and high-fidelity prototypes. More about how the about the prototypes and how they were made can be read in each respective section. Paper prototype in Section 6, the medium-fidelity non-functional prototype in Section 7 and the fully functional and high-fidelity prototype in Section 8.

4.4 Testing Methods

The usability of the first two prototypes was controlled and tested by user-tests[11]. These tests all followed the following format:

1. The interview subject are presented with the *Information Letter* from Appendix F. We go through together, and the user signs the document.
2. We introduce the research project with a written description of both the project and a rough description of the actual test. These can be found for each prototype in Appendix G.
3. We then present relevant background information to set the stage. Necessary technical knowledge to complete the test is also shared.
4. Finally we ask the subject to think out loud, to comment thoughts along the way, and we emphasize that it is the user interface we are testing, not the user's abilities.
5. The test has now started with prepared tasks and questions. These tasks are controlled, asked, and lead by a test manager. Notes are taken by the referent during the whole session. This is consistent with Statistics Norway's recommendations *[74] (Sundvoll* et al. ***2006*** *p.47)*.
6. At the end, there is room for further comments by the user, and some questions for clarification can be asked by the test manager and referent.
7. The test ends with the subject filling out a System Usability Scale (SUS) questionnaire so that the usability can be scored. More about this later.

Figure 5 shows that around half of the user selected for user-testing of a user interface should belong in the category of having limited IT-knowledge and interest, while also having limited domain knowledge. The people in this category are the ones that have the least likelihood of successfully navigating the interface, and they will thereby discover

[11] Due to time constraints, the third prototype was not tested. This is discussed further in Section 9.5

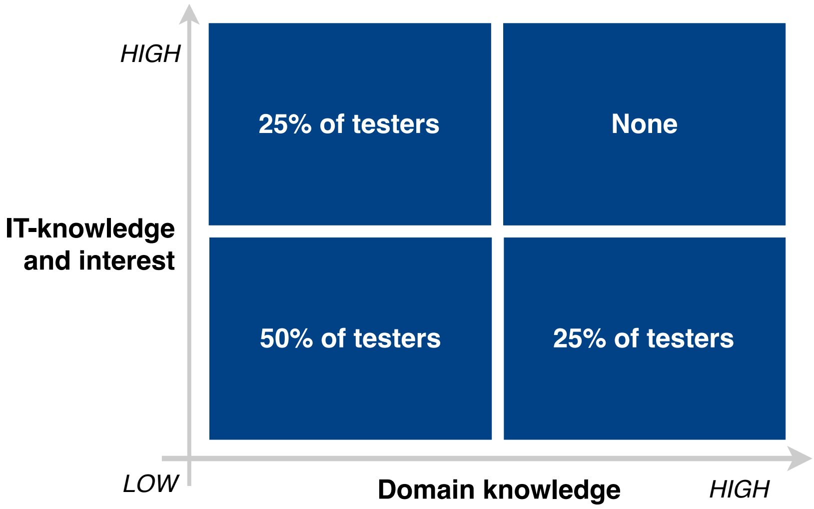


Figure 5: Types of users one should select for user-testing. Figure from *[85] (Toftøy-Andersen and Wold **2011** p.32)*.

most of the fundamental problems with the application. You should also test that the application communicates the conceptual model for both IT experts with little domain knowledge, and users with domain expertise and limited IT knowledge. This way you reveal errors that are domain or IT related. At no point are you testing expertise in both axes, because if the application works for the aforementioned groups separately, it will certainly work for the ones who are experts in both groups. The experts will only point out what they personally do not like with the application, rather than reveal real usability issues *[85] (Toftøy-Andersen and Wold **2011** p.32)*.

Number of Test Users

Literature states that having four users test a prototype is the most value when comparing cost and errors discovered *[86] (Nielsen and Landauer **1993**)*. Nielsen's later study showed that from doing four to five think out loud user-tests reveals 75% of the usability problems *[87] (Nielsen **1994**)*. He further specifically recommends no more than six to seven tests if the system is planned to have future iterations. Another source shows that approximately 80% of usability problems can be identified by running only five user-tests, according to the reported studies in Virzi's paper *[88] (Virzi **1992**)*.

These five user-tests is for testing the prototype for usability errors, and it assumes that the test is carried out without errors in the test itself. For testing the test, it is recommended to conduct what is known as a pilot user-test *[89] (Snyder **2003**)*. This is essentially the same user-test as you are going to do on the regular test users, but with the goal of quality ensuring the test for logical mistakes. These mistakes could ruin the test completely, or at least significantly lessen the quality of the results. The measure of the pilot-test allows the test plan to be updated, and by that ensures a smooth execution for later tests. It is stated that user-test professionals conduct two pilot-tests to be sure that all logical elements, flows of the prototype, and the script holds a sufficient quality *[85] (Toftøy-Andersen and Wold **2011**)*. Our prototypes were therefore tested with two pilot-tests in addition to five regular user-tests. Results regarding usability problems in the pilot-tests were recorded as regular results for comparison if no major test mistakes were discovered.

Recruitment of Test Users

When testing the paper prototype we wanted, according to Figure 5, to test subjects with low domain knowledge. Thus, focusing solely on the user interface, and not let the test users' domain knowledge come in the way. I order to ensure a relatively unbiased representative sample, we recruited students from different studies across different graduation years. Testing on people you know can yield overly positive results as acquaintances would often go further in an effort to be nice *[85] (Toftøy-Andersen and Wold **2011**)*. But, if one act professional you could have the benefit of getting more feedback as the users are less shy and more open to conversation. It could also be that some users are more honest in an effort to help their acquaintances as much as possible. Knowing that the more feedback they give, the better *[82] (Rubin and Chisnell **2008**)*.

In the following prototype, the non-functional prototype, we wanted to test the application on users with more domain knowledge. The recruitment was arranged through our company contact from Schlumberger. This gave us five users that fit the desired target. They were all informed of their rights regarding their personal information. All were also given the usual information of the project and the test, and in the end the SUS-questionnaire. Thereby, also testing on users with more domain knowledge than the previous iteration, but also as an update for the team at Schlumberger to show the progress that has been done so far.

System Usability Scale

The user experience for designing a configurable well montage should be as pleasant as possible. A way to achieve this is to aim for high usability. There are several guidelines as to how to achieve high usability, such as Jakob Nielsen's usability heuristics *[90] (Nielsen **1994**)*, Don Norman's fundamental principles for interaction *[91] (Norman **2013**)*, more about this in Section 4.5. To measure usability during and after a user-test, one often use what is called a System Usability Scale, or SUS for short. It was created by John Brooke in 1986, and measures usability by asking questions about effectiveness, efficiency, and satisfaction. Brooke describes it as a *"quick and dirty"* way to measure the usability of a system. It is often described as an industry standard, despite never have gone through any standardization process *[92] (Brooke* et al. ***1996**)*, *[93] (Brooke **2013**)*. The following are the ten questions that make up the SUS-questionnaire:

1. I think that I would like to use this system frequently
2. I found the system unnecessarily complex
3. I thought the system was easy to use
4. I think that I would need the support of a technical person to be able to use this

system

5. I found the various functions in this system were well integrated
6. I thought there was too much inconsistency in this system
7. I would imagine that most people would learn to use this system very quickly
8. I found the system very cumbersome to use
9. I felt very confident using the system
10. I needed to learn a lot of things before I could get going with this system

Each question is rated on a scale from 1 to 5, where 1 strongly disagrees and 5 strongly agrees. Then the usability is scored on a scale from 0-100. Despite its simplicity and few questions, it has been shown to prove quite an accurate way of measuring usability *[94] (Tullis and Stetson **2004**)*.

Privacy Concerns and Confidentiality

As this project will conduct interviews and test prototypes, some personal information needs to be stored. In order to comply with national law in Norway, regarding research with personal data, one must seek permission from the Norwegian Centre of Research Data (NSD)[12]. They receive applications and grant permission to conduct research that contains personal data. This project does not contain *sensitive* personal information, but it may store traceable data gathered during interviews or testing. An application was, therefore, sent and approved for both the interviews in the case study and the testing of the prototypes. The process involves filling out an online form and the creation of an information letter that are to inform the test-users of their rights. The letter created for the NSD application can be found under the name of *Information Letter* in Appendix F.

In addition, as the research of this thesis was done for Schlumberger, the research also had to follow a non-disclosure agreement by Schlumberger. This included, not sharing sensitive data for profit, or sharing of other company or business secrets from our cooperation with them. Before the thesis deadline, it will also be reviewed by Schlumberger's legal department to ensure these requirements are met. To make the research relevant to the scientific community, the research is presented in the most open way possible.

4.5 Interaction Design

*"Today, and even more so in the future, a poor user experience might become a serious issue and undermine the reputation of the software" [95] (Esposito **2016**).* In order to successfully facilitate the data that comes from information overload in DrillPlan, into something usable for the user, we have to look at how to design a competent user interface with appropriate *interaction design*. This is a complex field which involves everything related to designing interactive products. Interaction design can be described as multidisciplinary, as illustrated

[12] https://nsd.no/nsd/english/index.html

by Figure 6. The figure illustrates some of the different aspects that have to be taken into account when creating an interactive product.

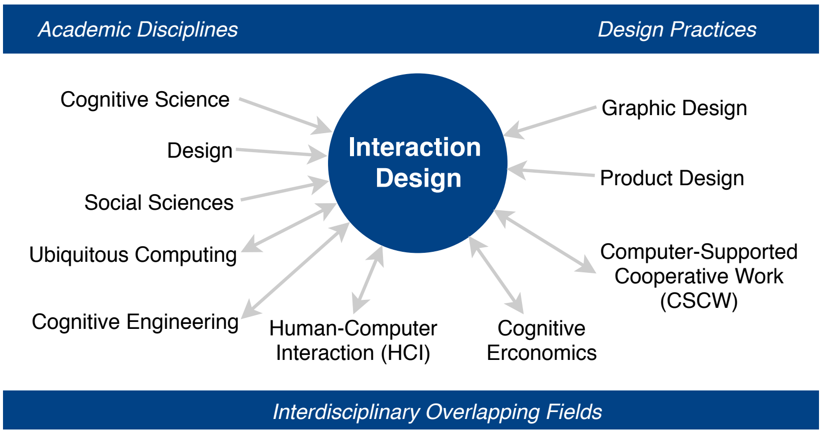


Figure 6: The multidisciplinary nature of interaction design *[78] (Preece, Rogers, and Sharp **2015**)*.

The main goal of an interactive product is that the user should at all times know what they are doing, and how to proceed. Its focus should, therefore, be *"to enhance people's understanding of what can be done, what is happening, and what has just occurred."[91] (Norman **2013** Chapter 1)*. To ensure that the design enhances the product and that its function is communicated clearly to the user, one needs principles and guidelines to follow. Don Norman defines five principles for interaction: *feedback*, *affordance*, *signifiers*, *mappings*, and *constraints [91] (Norman **2013**)*.

- *Feedback*: Communicates a reaction of an action, meaning, the user know what has just occurred by being given an appropriate reaction.
- *Affordance*: What interactions are possible to enact. The objective truth of possible actions.
- *Signifiers*: The perceivable part of the affordance, which helps to close the gap of what the user can do. The perceivable truth of possible actions.
- *Mapping*: The relation between two sets of elements, with regards to how they are interlinked.
- *Constraints*: Restricts certain parts, to enhance the perceivability of others.

These principles are based on what an action entails, which Norman describes as the seven-stage model of an action, which can be seen in Figure 7. By splitting an action into several steps, one can more clearly see what is important for an action to occur, as understanding the action itself is vital when guiding your design. Thus, by concretizing the action that the user should do in a program, with the aid of the five principles, one has clearer ways of identifying potential problems with the design *[91] (Norman **2013**)*

In addition to the five principles of design, Norman describes the notion of the *conceptual model* as *"perhaps the most important"* principle of all. The conceptual model is a de-

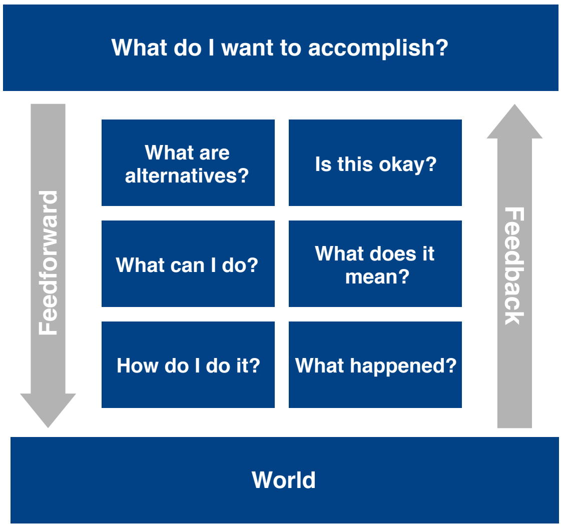


Figure 7: The seven steps of an action, from *[91] (Norman **2013** Chapter 2)*

scription of how the system operates and how its concepts relate to each other. Meaning, through metaphors and analogies that are deployed in the design, the actual system and its interaction should become clear to the user *[91] (Norman **2013**)*. This could be seen as the foundation of a system. Thus, supporting every decision that will be made, as it is a *idealized* view of how the system works. The conceptual model is *"not the user interface"*, but the concept of the system in a high-level manner *[96] (Johnson and Henderson **2002**)*.

The conceptual model is what the designers have in mind when creating a system, with regards to how it operates and is organized with the aid of proper use of metaphors and analogies. These concepts, as perceived by the user, is what is called the *mental model*. It does not always align with the conceptual model, and different users with different backgrounds may possess different mental models of the same system. To reach a better understanding, and thus better usability and interactions with the system, the designers should strive for making the conceptual and mental model correlate as closely as possible. If these two correlate closely, there will be fewer misunderstandings and misconceptions for the users, and thus increase the usability of the system *[91] (Norman **2013**)*.

> *There has been an underlying assumption that because humans are so inherently flexible and adaptable, it is easier to let them adapt themselves to the machine, rather than vice versa.*
>
> Handbook of Usability Testing
> *[82] (Rubin and Chisnell **2008**)*

Jakob Nielsen defines ten heuristics for user interface design, which can be used as *"broad rules of thumb and not specific usability guidelines"*, thus providing some supplementary instructions for when designing for usability *[90] (Nielsen **1994**)*. These can easily be combined with the aforementioned goals and principles for additional guidance when making a competent design with a focus on usability. Examples of some of these are, consistency, error prevention, and recognition rather than recall, to name a few. By including more than one set of guidelines and principles, one has more support for assessing the design, which could minimize the likelihood of bad design and make the identification of user's problems easier.

It is important that the information and data in a system are presented properly. When the complexity of the data that is to be shown increases, consequently, there is a need for more advanced user interfaces. It should support what Shneiderman describes as a "Visual Information-Seeking Mantra", to avoid the potential for information overload. It entails structuring the data in the series of events: overview first, zoom and filter, then details on demand. By applying these tasks into the user interface, the type of data and its hierarchy of complexity is more easily projected to the user, thus making the user interface more compliant in avoiding the potential for information overload *[97] (Shneiderman **1996**)*.

Successfully complying with the design principles and the seven-stage model will not do much on their own if the product is not regarded as usable. Thus, one should take usability into consideration when designing. Preece defines the following usability goals: *effectiveness*, *efficiency*, *safety*, *utility*, *learnability* and *memorability [78] (Preece, Rogers, and Sharp **2015**)*. A product should by these goals be; easy to learn, its functionality rememberable during use, while still being efficient in what it does, and to do it safely and effectively. The challenge here is not understanding the importance and viability of each goal, as they are self-explanatory with regards to usability. The challenge for the designer is rather how to create an elegant solution, while also making sure each of these goals is reached in a satisfactory manner.

4.6 User-Centered Design

Jakob Nielsen writes that *"(...) even the best usability experts cannot design perfect user interfaces in a single attempt." [98] (Nielsen **1993**)*. What a designer might think is a good design, might come from that the designers having gained too much emotional attachment to a project, without the proper evaluation in the process of designing. Which typically results in the conceptual model not communicating properly to the users. An approach for handling this issue is to design the user-interface in user-tested iterations. This allows the interface to be continually tested and improved without the designers gaining a too large of an emotional connection to the product. The way forward is therefore evaluated by the users, not the designer. Nielsen shows in the article that by designing iteratively based on user-tests between each iteration, greatly improves the usability of the user interface. The idea of including the user in the design process is called User-Centered Design (UCD) *[98] (Nielsen **1993**)*. The iterative work, combined with observations, testing, and retrospective analysis ensures that the design converges towards a functional solution. This is the basis of what Norman calls *design thinking [91] (Norman **2013**)*.

> *UCD is an evolutionary process whereby the final product is shaped over time. It requires designers to take the attitude that the optimum design is acquired through a process of trial and error, discovery, and refinement. Assumptions about how to proceed remain assumptions and are not cast in concrete until evaluated with the end user.*
>
> Handbook of Usability Testing
> *[82] (Rubin and Chisnell **2008**)*

What methods and guidelines typically implemented while doing user-centered design greatly varies. One should try to apply clearer guidelines, and integrate those into the design process, to avoid the ambiguity that might arise while iterating. The degree of commitment, however, to the idea of UCD and potential UCD-guidelines vary, as these are implemented with regards to cost versus benefit. What is important is that the user is involved in the entirety of the process. Making the user's involvement always support the development of the design *[99] (Gulliksen, Göransson, Boivie,* et al. ***2003**).*

As mentioned, the degree of implementation of UCD often comes down to cost versus benefit *[99] (Gulliksen, Göransson, Boivie,* et al. ***2003**).* Added to the fact that both the usability and the actual effectiveness of UCD is hard to measure, it could be seen as a less important design process as its impact is elusive. The cost of implementing UCD can exceed as much as 10 percent of the project's budget. However, studies have shown that UCD is generally worth it despite its high cost. The point of interest is *how* one implements the different aspects of UCD into the development, for it to make its impact on the design. Consequently, this is why UCD is a popular concept to implement, for its effectiveness in making usable designs with the aid of the users' involvement *[100] (Mao, Vredenburg, Smith,* et al. ***2005**).*

5 Case Study

This section will describe the first steps in tackling the project. It describes the phase of information gathering in order to better understand the concept of a one-page summary and how DrillPlan's solution can be improved. This was part of the preliminary research, conducted in the fall of 2018 *[75] (Einarson and Fredriksen **2018**)*.

After receiving the master's thesis proposal, discussions with Schlumberger and within were undertaken to figure out how to proceed. Schlumberger inquired an open and exploratory solution for their problem, and that no boundaries should be set in terms of scope or potential usage of third-party software integration. The proposal was used as a guideline for the project, and further work and research was needed in order to proceed. Schlumberger provided documents regarding how their current implementation of Well Montage was made, and access to DrillPlan as to further enhance the insight into how it is used.

Little to no research has been conducted on mapping the principles and requirements of a one-page summary. According to Sadhwani et al. a one-page summary should at least contain:

> *"(...) Casing design and specifications, Formation evaluation, MudLogging, Bit specifications, Hydraulics, Casing design and specifications, Drilling Fluid properties, Cementing design, BHA's design*[13], *Wellhead/BOP's assemblies*[14], *Geological information, Offset Well Analysis including the drilling events associated and a General Comment section with Lessons Learned and additional measures to be taken into account during the drilling phase." [69] (Sadhwani, Sanderson, MacFarlane,* et al. ***2016****).*

Apart from that, there is little to no information published regarding commonalities and procedures as to how a one-page summary is made and used within the well planning phase.

5.1 Interviews

In order to gain insight into the use of the one-page summary, interviews with people that have experience in the field had to be conducted. Initially requested to Schlumberger were users of DrillPlan familiar with the use of Well Montage and also the one-page summary in general. However, seeing as DrillPlan is not fully adopted, and thereby had a limited set of veteran users and with sales negotiations still left up on the table, made this difficult to realize. One would prefer not to risk that the limited seniority one or two master students could interfere with ongoing business negotiations. A compromise was made with the solution to conduct interviews with three different employees working within the field of customer relation of DrillPlan in Schlumberger. They were all located in different parts of the world and had, therefore, different insights and experiences regarding the one-page

[13] BHA=Bottom Hole Assembly
[14] BOP=Blow Out Prevention

summary's use in their region. Each interview contained the same outline of questions. These questions aimed to; assess the one-page summary's purpose and use, describe the average user, and to determine what possible enhancements could be made to today's solution in DrillPlan. The interviews followed a semi-structured flow, thus leading to a more natural conversation with the subjects. The interview guide that was used can be seen in Appendix B.

Results From Interviews

The one-page summary was through the interviews ascertained as an *"open secret"*. Its use was prominent in the later stages of well planning, where overviews of the situation are necessary and all the final decisions have to be made. One interviewee said *"Everybody is using one!"* regarding its use in well planning. As it is used so prominently, and usually requires manual labor to construct, the interviewees concluded this as to why the users had been requesting it to be implemented into DrillPlan. Through these interviews, three distinct use cases emerged:

- *First use case*: On smaller onshore wells with low volume and lower risks, the one-page summary plays the role as a *"recipe"*. Meaning, it is used as the main reference point when drilling. *"They print out the one-page, and put it up at the well site, and use it as their only reference when drilling"*, one of the interviewees explained *[75] (Einarson and Fredriksen **2018**)*. These are simpler wells, which entails that they are often drilled straight down without any curvature. In addition, many wells with the same parameters are often drilled in close proximity to each other. Thereby reducing the need for explicit planning for each well, and one can reuse the one-page summary as a recipe for several wells. Regarding this, the interviewee described that companies could share this recipe, and *"they will edit it only if something goes wrong, if not, it stays the same for several wells" [75] (Einarson and Fredriksen **2018**)*.
- *Second use case*: On larger wells, often offshore, the use case is entirely different. Instead of planning the well for a few days, and reusing parameters that have been used previously - the planning takes months or even years to complete. In this use case, the one-page summary will not suffice as the main reference point, and the entire drilling program is needed. However, as larger drilling programs can be up to hundreds of pages, the compact synopsis of the well is more easily tenable in discussions leading up to the drilling. *"Here they operate with a presentation of around 50 slides, going in depth of different parameters, having the one-page summary as an even shorter synopsis"*, one of the interviewees explained *[75] (Einarson and Fredriksen **2018**)*. After the planning is over a one-page summary may, however, be hung up on the wall of the rig to serve as a general overview of the site if need be.
- *Third use case*: The third use case was to use the Well Montage as a basis for application forms for local jurisdictional purposes. E.g. users from Canada described formatting or copying and pasting fields from the one-page summary, onto the submission form, thereby rendering the Well Montage as a tool for helping with

jurisdictional applications.

Additional information emerged during the creation of the aforementioned use cases. The average user of DrillPlan is quite comfortable with digital user interfaces. *"They usually sit in front of the computer large parts of the day, and are therefore comfortable with using different kinds of software"*, one interviewee mentioned *[75] (Einarson and Fredriksen **2018**)*. However, workers on the rig itself might be lacking experience with computers, and even having limited Internet access.

With regards to the limitations of the current Well Montage, there were some. At the moment, you have to regenerate the Well Montage every time an update is made in DrillPlan since the Well Montage is a static Excel sheet. This is in stark contrast to DrillPlan's otherwise dynamic interface. The users did, however, like that it was based in Excel, since this is something they are both used to and comfortable with using. In addition, a document in Excel meant making it easier to modify before printing the Well Montage, which was something that was often done. Other users were discussing the formatting and found it not visually pleasing enough. Meaning, for some meetings a more technical and data-heavy Well Montage can be used, but other times there is a need for a more graphically pleasing version. One of the interviewees recounted a meeting with a customer where they were discussing the layout of an unrelated well report:

> *What is this font? It's too small, and doesn't look good. And the colors on this chart? You have only used primary colors, and it looks boring. I want better colors when I show this to my superiors, cause this report looks quite boring as it is now.*
>
> - DrillPlan customer
> Paraphrased from interview
> *[75] (Einarson and Fredriksen **2018**)*

The quote describes the fact that the users expect a certain degree of visual flair when looking at reports. *"The more graphical it gets - the better"*, mentioned one interviewee, as some customers have different interests with regards to how a report should look. However, as the one-page summary is a quite data-heavy report, in some cases it should stay that way. The interviewee further elaborated *"maybe in a meeting you would like a more graphically pleasing paper, but at the well itself you only care about the numbers" [75] (Einarson and Fredriksen **2018**)*.

Despite this, most users concluded that Well Montage, in its current implementation, all though limited and cumbersome, was all in all adequate. However, more in-depth functionality and modifiability were certainly wanted and needed by the customers the interviewees represented.

Through the interviews, it became apparent that the use of a one-page summary is widespread. Despite this, there are few commonalities or collaborative efforts done in the oil industry regarding it. It might be of interest to propose a common standard, but standardization is difficult and costly to achieve especially with regards to the conservative silo-mentality situated in the industry today *[101] (Ershaghi, Omoregie,* et al. ***2005****)*. However, the one-page summary's purpose seems clear. The one-page summary is most often used as a catalyst for knowledge creation and knowledge sharing. This is done by orchestrating large amounts of data onto a summary which is more malleable *[25] (Monteiro, Pollock, Hanseth,* et al. ***2013****)*. For it to be as effective as possible, it is important for the user to know its layout and formatting. By increasing its readability with more visual cues, its purpose can be enhanced, and thereby having a greater effect on the user *[26] (Nygren and Henriksson* ***1992****)*.

5.2 Proposed Initiatives

Before proceeding with the proposed initiatives, the following three lessons from the interviews and our research was crucial to how these were designed. Firstly, DrillPlan is already a very collaborative platform, which is a feature that is unused in today's Well Montage. By including some collaborative features into an improved implementation of Well Montage, the collaborative nature of DrillPlan would not be overlooked. This would support knowledge creation and sharing, e.g. through live updates. Secondly, the use of Excel as a recognizable format. It was stated that it was not a need for it to be based on Excel, but for backward compatibility reasons, support for Excel should be prioritized. Lastly, the possibility of using third-party applications to aid the development was suggested by the team at Schlumberger. This meant that not all of the proposed solutions had to be feasible as in-house development.

The next four initiatives make the basis for the proposed solution. These were based on the findings from the preliminary research and lay the foundation for the continuation of the project.

- *Templates*: By providing several templates with premade setup of the Well Montage, the user would no longer need to edit an Excel sheet to fit their needs. Instead, they can choose the most fitting template, by filtering what needs they have for the template they are after. This feature - the template browser - is illustrated in Figure 8. For this feature to reach its potential, one has to include enough templates to cover most of the identified use-cases. The user can thereby find a fitting template for their needs without any additional customization, which supports users less proficient in technology and users with fewer intentions of creating customized Well Montages.
- *Modules*: Users might be demanding several templates for several different use cases, which would impose that Schlumberger always has to stay on top of user demands of new templates. To counter this problem, each template is built upon the idea of modules[15]. These modules function as columns with data fields populated by different well planning specific categories. By making the templates customizable by adding or removing modules, each user can then customize the Well Montage for

[15] Modules were originally named widget. To encompass a more fitting name, widgets were in this report renamed to modules. The drawings were made before this change.

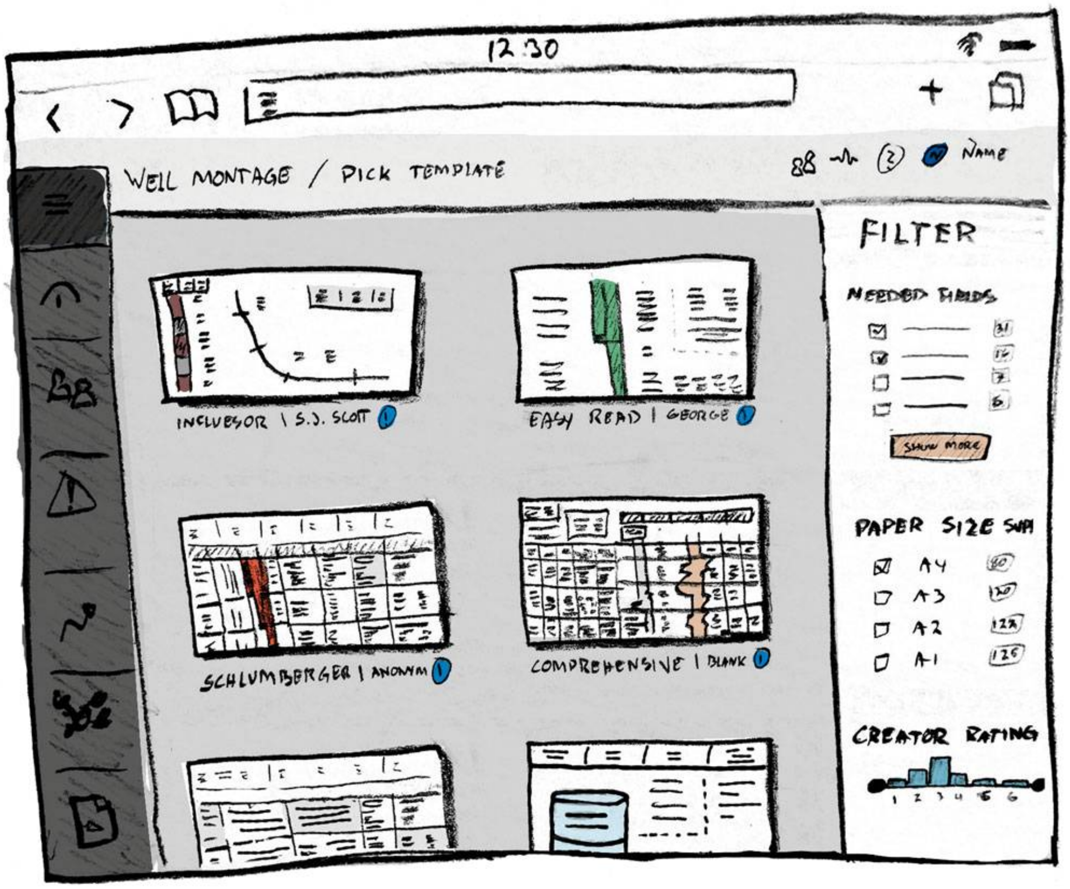


Figure 8: Drawing of the functionalities of the template browser.

their personal needs. It is important that this feature, as seen in Figure 9, is user-friendly. By focusing on the user-friendliness of this feature, it will encourage the user to customize the Well Montage, if need be, instead of scaring them back to using Excel for customizing their Well Montage. It is important that the users' mental model match the conceptual model of the system so that the concept of templates being built up by modules is to be communicated clearly *[91] (Norman* ***2013****)*.

- *Custom modules*: For even more advanced users, another initiative is to customize the modules themselves. Meaning, not being limited to the proposed modules, but having the possibility of further customize the look and contents of each module through code. An example of how this can be done is seen in Figure 10. This has the potential of being a security risk as queries can be done directly towards the database for when coding each custom module. Whether each module is re-programmable or features substantial customizability options, the idea of customizing each module is something that should be evaluated further.
- *Interactive Well Montage*: In addition to supporting the aforementioned backward compatibility of printable documents and the use of Excel, an additional initiative is to use the power of interactive web interfaces. Meaning, why limit Well Montage to

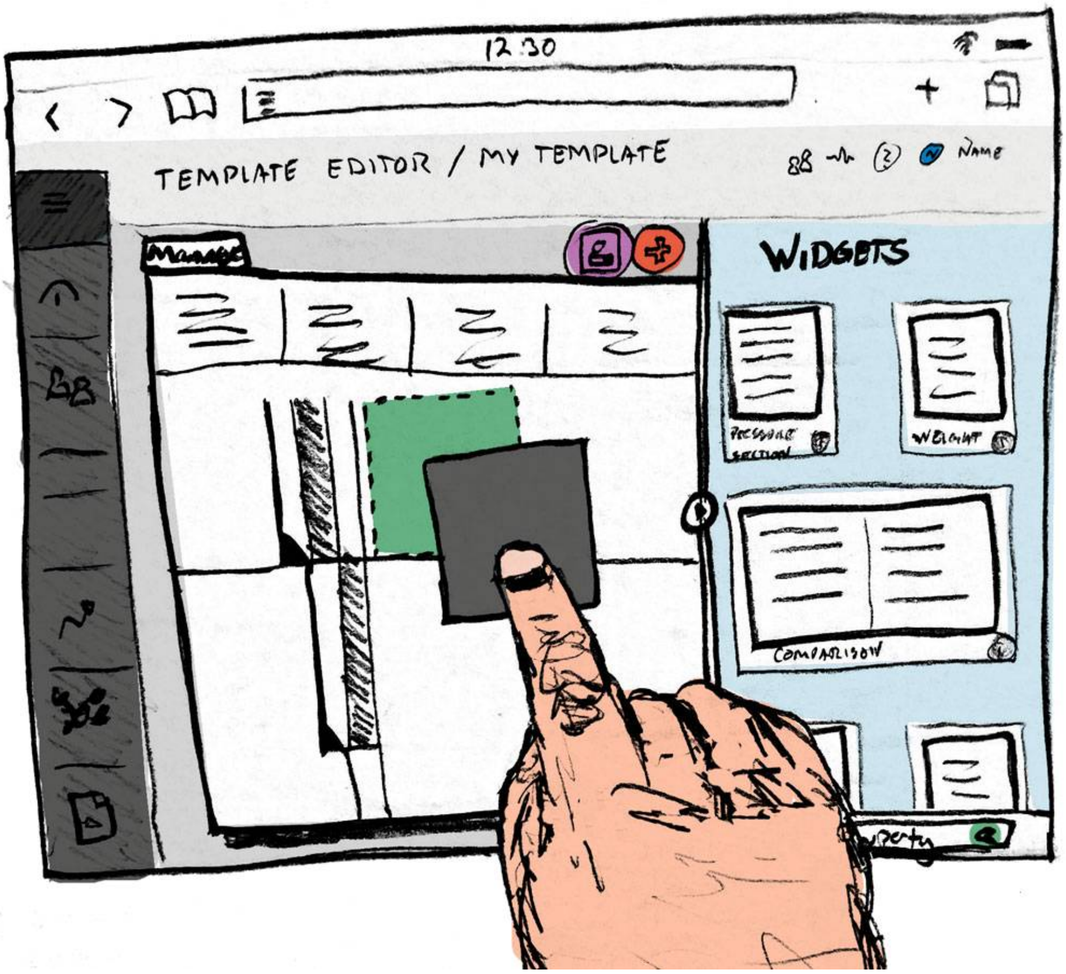


Figure 9: Drag and drop new modules to customize the Well Montage.

a digital copy of a static non-interactive document like Excel? An example of this can be seen in Figure 11. In the figure, the user is hovering over specific fields to access additional information. This could be seen as future-proofing Well Montage for when internet access is readily available everywhere, and the use of tablets and smart-phones is more common on the well site itself.

Proposed Solution

Our proposed solution of how to improve Well Montage was a mix of the four described initiatives. The user's journey starts by picking a template. This template is interactive, so e.g. potential additional data and version control is shown. Thereby following Shneiderman's tasks of achieving proper data visualization; first giving the user an overview through the templates, then, depending on the user, supporting several different tasks as to how to filter and extract the user's preferred information *[97] (Shneiderman **1996**)*. Each template should also correlate to the chosen paper size, to ensure printability. It should be made clear that each template is built up by the use of modules, and that each module is customizable, to encourage the user to customize the template if they want or

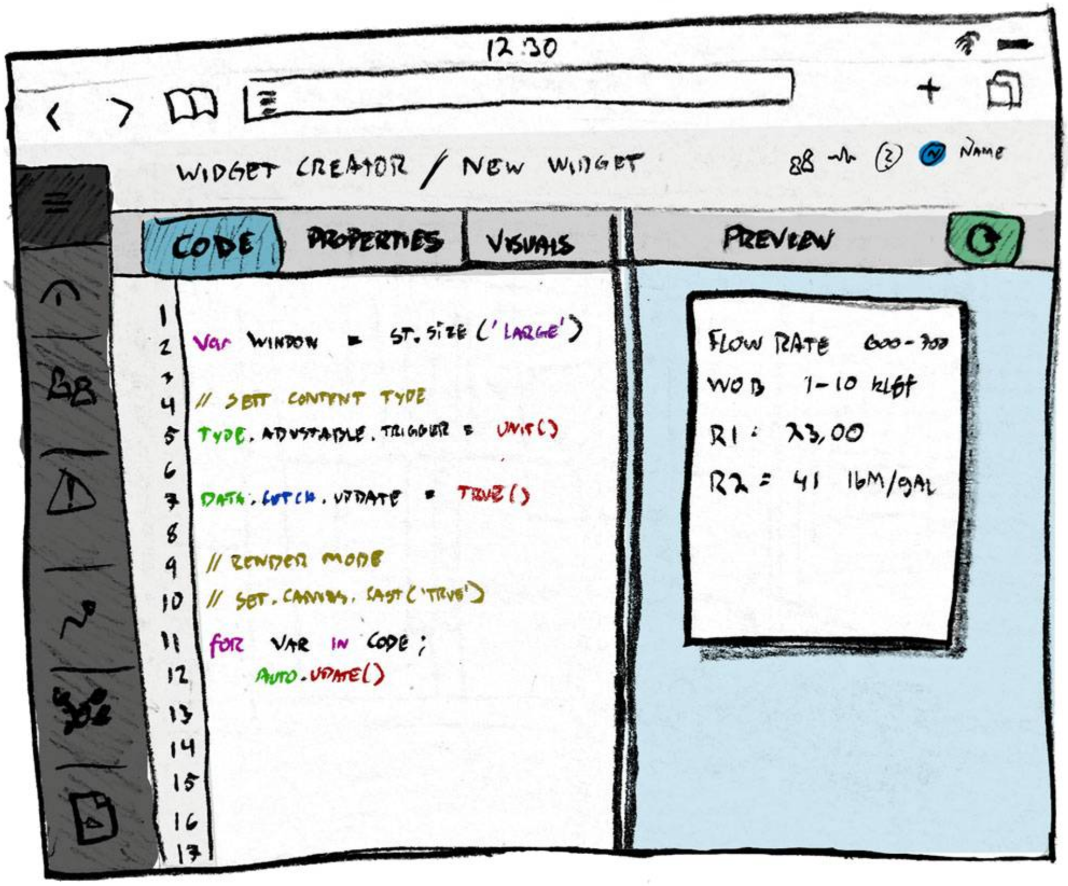


Figure 10: Customize modules through a built in code editor

need to. The Well Montage could then be shared to any one of interest. That being said, it was important to avoid feature creep when going forward, and thorough testing and prioritizing of requirements should be done so to prioritize the most important features.

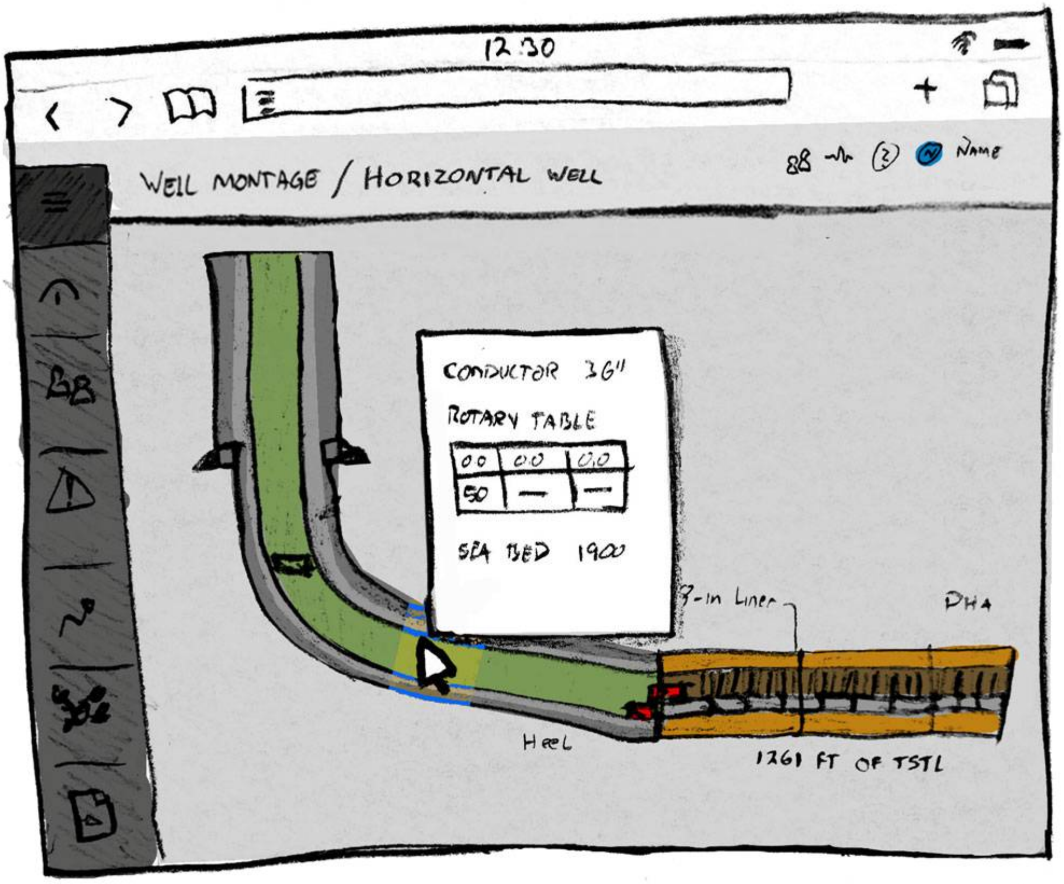


Figure 11: Drawing of an interactive Well Montage

6 Paper Prototype

This section will describe the process of establishing requirements, the creation of the paper prototype, and its evaluation through user-testing. This prototype, and the subsequent prototypes, is based on the proposed initiatives as described in Section 5.2.

6.1 Requirements Engineering

Going from *specialization* to *operationalization* in a project, requires a set of prioritized requirements. From the proposed initiatives in the case study, with the aid of methods from Knapp's Sprint book, requirements engineering could take place.

"Be Able To" Technique

One technique we used in the early stages of this project was an adaptation of *"How Might We"* from Knapp's Sprint book. This technique was originally intended to be used during the conversation with domain experts. The team is supposed to write their ideas on sticky notes on the format of *"How Might We"* followed by the problem your idea solves. This is a way of storing the ideas as they come up, on a format which is easily communicated. We, on the other hand, did similar research and interviews in the case study, with regular notes taken along the way. Inspired by the technique, and with a desire to have our ideas in the same format, we, therefore, took 15 minutes of writing on sticky notes to kick-start this phase of the project. The exercise helped us see what ideas we as a team of two already agreed upon. The notes were created from both memory and writings done last semester. The sticky notes were written in the form *"Be Able To"* + the idea. These sticky notes were not limited to any particular perspective and could be in anything from a user's point of view to a developer's mindset, or in the context of a business owner.

Similar to what is described in Knapp's Sprint book, our sticky notes were after the writing session put on a white-board and presented when hung up. As a team effort, they were then organized into different groups of similar category. Naming these groups or themes becomes obvious later on when a sizable grouping of the notes had emerged. This technique made it easier to get a grasp of all the ideas. The result of this session showing our organized sticky notes from the *"Be Able To"* exercise can be found in Figure 32 in Appendix D.

The next step in Knapp's book is to prioritize the identified *"Be Able To"* sticky notes. We used an adaptation of the *planning poker technique*, well-known in the software development industry for both prioritizing in terms of importance and for giving an hour estimation of development time. Firstly, the sticky notes were digitized into a table editor. Then both team members sat down individually for a few minutes noting down the importance of each requirement on a scale from 1-5, with 5 being the highest. While doing this, we also estimated a development effort for each task noted in hours. These numbers were then presented to each other for each requirement, which led to a small discussion on that topic, clarifying both the item's description and its goal. By the end, we had a prioritized list sorted by importance with a shared estimation of development hours. Part of this exercise's intention is to have small discussions that lead the team members to have a better common understanding of the challenges and objectives of the project. Also, by doing the prioritization and estimation individually, one avoids the cognitive bias of anchoring; where people get too influenced by the initial information given. The list of prioritized requirements can be viewed in full in Table 9 in Appendix D.

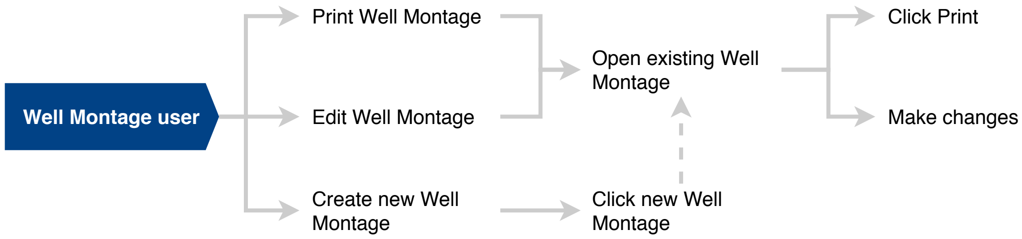


Figure 12: High-level map of interaction for a Well Montage user

User Maps

A useful exercise for finding what elements to test in a prototype is to create a map illustrating typical use cases *[76] (Knapp, Zeratsky, and Kowitz* ***2016****)*. Figure 12 shows a limited high-level map of user interactions for our intended system. It shows three of the main activities that the application is supposed to support from its start screen - in compliance with the prioritized requirements established in Appendix D. The first activity, *"Print Well Montage"*, and the second one, *"Edit Well Montage"*, both require the user to open the appropriate Well Montage to complete the action. The last activity, *creation of a new Well Montage*, however, would require to click a button at the start screen called *"New Well Montage"*. When creating this map, we had to boil down all the requirements into what serves as an absolute minimum viable product. Meaning, what does the application *has* to include to serve its purpose.

From the creation of this map, it was apparent that the activity that had the most interactions was the aspect of *"Make changes"*. It also had the most uncertainties as to how well interactions would communicate, which consequently would mean the most rework if done wrong. It stood out as a feature that had the most options and difficult decisions in general. The other interactions in the map seemed more basic and would, therefore, be tested as part of a larger test of flow in a later more feature complete prototypes.

To further explore what *"Make changes"* should include, we created a new map describing some of the most important ways the user can make changes to the Well Montage. The map can be found as Figure 13. The actions listed in this map naturally belongs to a category of similar actions. That is why the actions funnel to a specific category that the user should seek in order to make the change. The figure shows a distinction between edits done; to the entire page, to specific modules, and to individual attributes of each module. These three categories of actions all have to be controlled from an intuitive interface, to increase the chance of communicating the conceptual model to the user. The map helped to identify these categories.

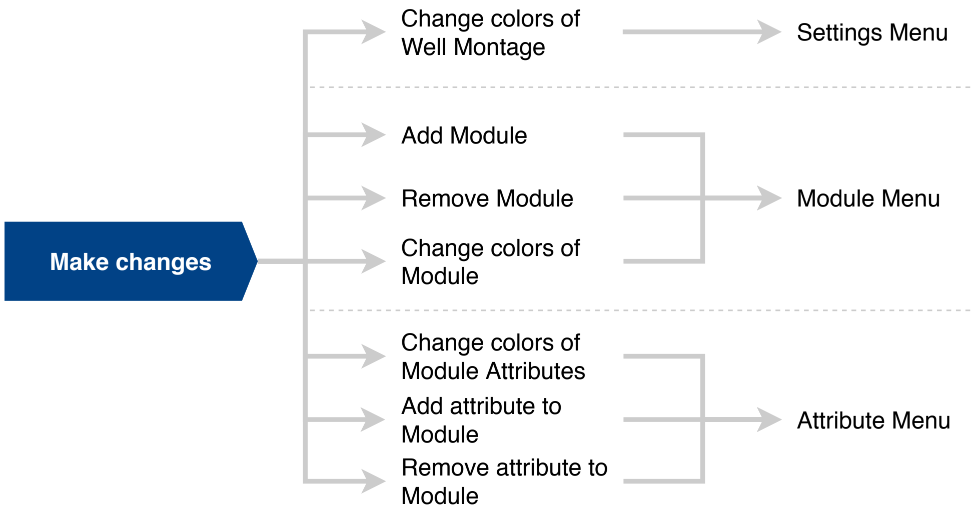


Figure 13: Lower level map of important changing actions inside the Well Montage

Core Functionality

The three categories in Figure 13 are what we consider as core functionality for the application suitable for paper prototype testing. These have in common that they involve decisions for the visual design in reaching the same goal, while also having a high risk of being wrongly implemented. Wrong in the sense of not meeting the users' expectations in terms of intuitiveness and usability, and thereby fail in establishing the correct mental model. It is prioritized for early testing as it is a high risk to the project due to the high cost of developing a new interface for each of these functions. Modern development methods, like agile, often aims to minimize risk by being risk-oriented in the decision making, and the exercise of creating a user map, therefore, helped us to pinpoint a specific focus area for this prototype to minimize risks.

Knapp's Sprint book advocates that the next step in the creative prototype process is to draw quick drafts for each of the functionality that is to be tested. It is important that each of the drafts is created as independent from another as possible. This to increase the chance of capturing all creativity in the group before the opinions of others influence the designs. The session lasted around ten minutes for each of the three categories in Figure 13. This spawned an average of around three to four drafts for each team member, illustrating different user interfaces. These were later compared and discussed.

During discussions of the quick drafts for interfaces, we needed a reference for standard designs. In order to get a better feel for how to design the different elements and to further the inspiration of the possibilities, we had a session where we gather bits and components from commercially available applications. When designing a new user interface it is important to follow certain unwritten rules as to how elements should look and interact. A new user interface should include familiar elements and ways of interaction for its users. The document made for inspiration can be found in Appendix E.

6.2 Prototype Description

Based on the drafts from the last exercises, combined with the aforementioned inspiration document, we could now begin creating the actual paper prototype that would be tested. Some of the drafts were similar enough to be merged into a singular design, and some had interesting elements that could be adopted by other designs. After discussion, we ended up with a strategy of creating two independent paper prototypes that were to be tested on the same set of users. This way we could try out, and test, more ideas than what would be possible with having only one prototype. By testing two individual prototypes, we hoped that users could effortlessly analyze through comparison rather than answering difficult questions regarding the design. They would thus be able to provide more effective and meaningful feedback. We could then ask them to respond to which prototype they preferred, and through the justification of their choices, get more useful feedback than we otherwise would with only one paper prototype.

The two paper prototypes aim to be easily separated and recognizable simply based on the way the interaction takes place in each of them. Paper prototype 1 is the more touch screen inspired of the two. It implements drag and drop functionality. It features dynamic, context-sensitive panels instead of separate windows, and it is in total the more modern approach. It is inspired by creative tools such as draw.io[16] and Marvel[17] App in both the way it operates and how menus appear. Meaning, for example when a user clicks a table element in draw.io, a menu for adjusting specific parameters for this element appears.

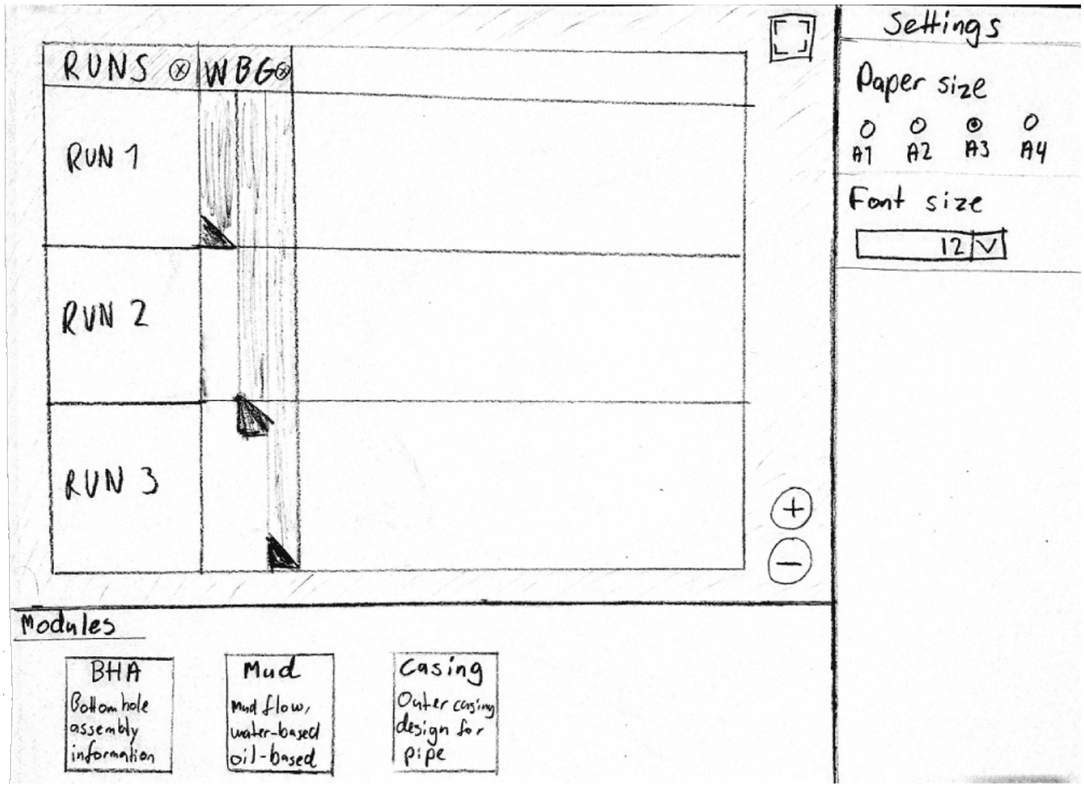
The second prototype is more traditional in the way it handles multiple list objects and how additional content is shown. The philosophy here is to have windows appear on top of the view, rather than having them slide in as a context-sensitive side-menu, as in paper prototype 1. This makes the application less cluttered, and this as a consequence makes clearer barriers as to where edits can be done and not. This approach also has the possibility to handles more information better, as screen real estate is better utilized.

Main Screen

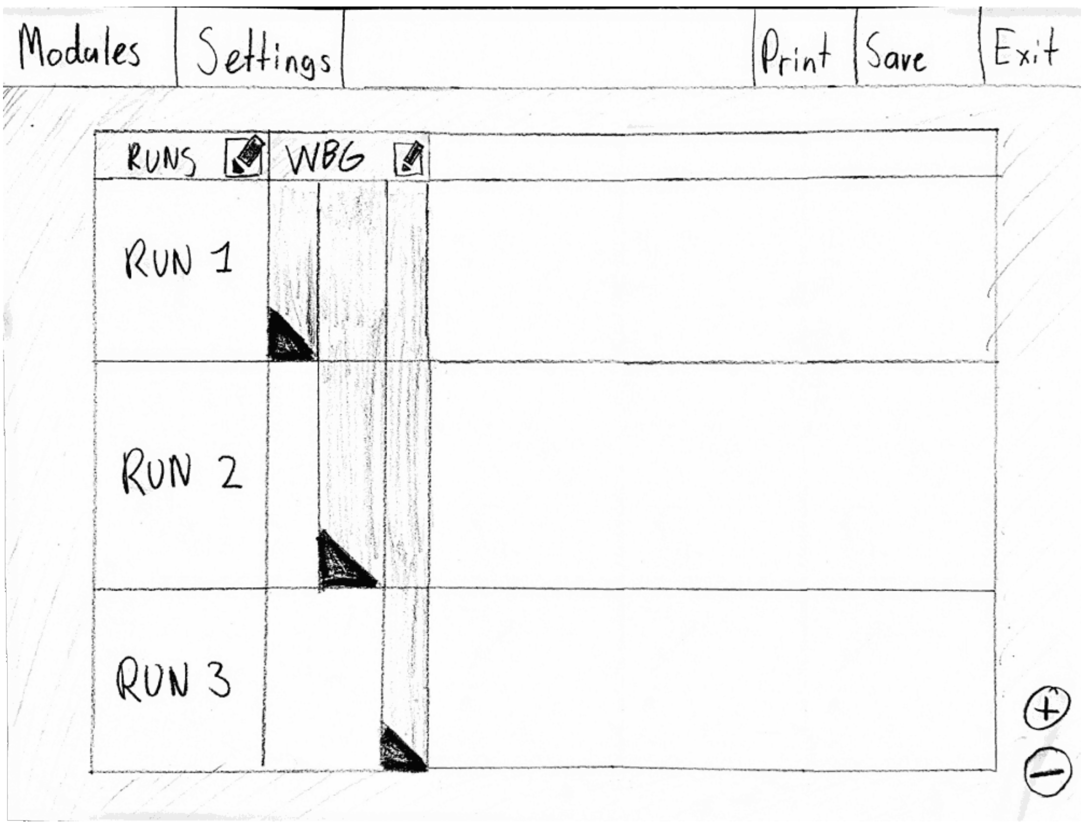
Figure 14 shows the first initial screen in both paper prototypes. Both prototypes include a workspace for designing the Well Montage, which is called *canvas*. We chose to include two already placed modules on the canvas for both prototypes, namely the *"Run"* and *"WBG"* modules. This was done in order to make the initial screen less abstract and more in line of how a Well Montage currently look. We also thought that a blank canvas could seem more daunting for new users and that the domain and setting for the test would be more difficult to explain. Also common for both prototypes was the inclusion of two zoom buttons located at the bottom right corner of the canvas.

From Figure 14 one can see that paper prototype 1 is divided into three panels. Taking up most of the space of the application is the canvas - spanning from the upper left corner diagonally down to about 3/4 of the screen. Settings for the entire page is located on the

[16] https://www.draw.io/
[17] https://marvelapp.com/



(1) Paper prototype 1



(2) Paper prototype 2

Figure 14: Home screens of the paper prototypes with the modules *"Runs"* and *"WBG"* preloaded

right side as a sidebar - spanning from the top to the bottom of the screen. This panel changes from global settings when nothing is selected - to module specific setting if a module is selected, more on that later. On the bottom of the screen, there is a panel with the available modules. The available modules here is *"BHA"*, *"Mud"*, and *"Casing"*.

Paper prototype 2 has a more traditional designed menu bar located on the top of the screen. This menu bar is were the user access options for *"Modules"*, *"Settings"*, *"Print"*, *"Save"*, and *"Exit"*. The *"Modules"* and *"Settings"* buttons will both open a new window where the relevant options are listed. As for the first paper prototype; *"Settings"* is where you control settings for the entire Well Montage, and the design of this window can be seen in Figure 16. *"Modules"* is where you move, add and remove modules. The rest of the page is dedicated for the Well Montage, i.e. the canvas. The canvas in the second paper prototype allowed for no *Drag and Drop* features. Any edits or changes must be made, without exception, in a separate window.

Modules

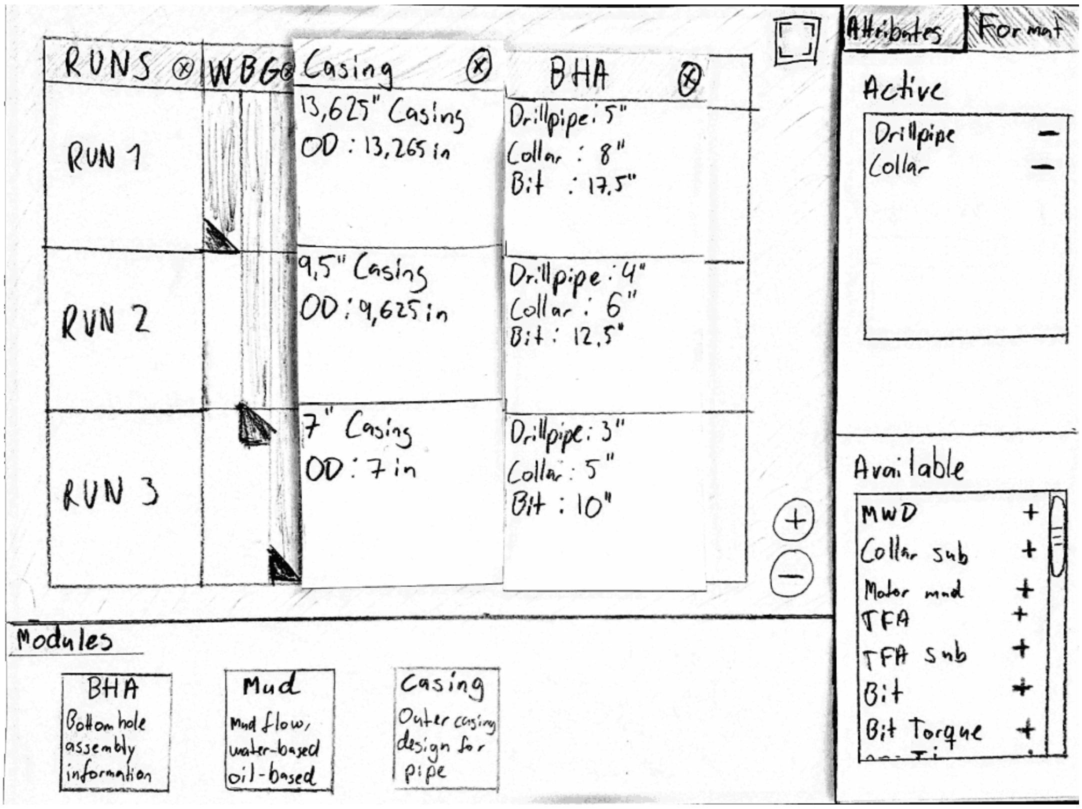
To add new modules to the canvas in the first paper prototype, users can click the desired modules from the *"Modules"* panel. Dragging and dropping it to the canvas will also work. By utilizing the drag and drop functionality, you will get the benefit of being able to freely place the module while getting a preview of the module's placement. The other modules should dynamically move according to the user's input until the release of the dragged module. To move already placed modules, the user has to click and drag on the title of each module. To remove a specific module, you click the x button at the top right corner of each module.

Adding modules to the second paper prototype is done through the window appearing after a click on the *"Modules"* button. A window will appear on top of the canvas, partially showing the canvas behind. The *"Available"* modules one can add are shown as a list and next to it is a list with the modules that currently exist on the canvas. To successfully add one of the available modules one selects the desired module from the *"Available"* list and use the arrow buttons to transfer it to the *"Active"* list. Multiple modules can be added simultaneously this way. These edits continuously change the canvas underneath, and one can see the changes happening behind the window. If one would like to change the order of which the modules appear in, the user has to navigate to this exact menu, as this is the only place to do so. This is also the only menu where you can remove modules from the canvas.

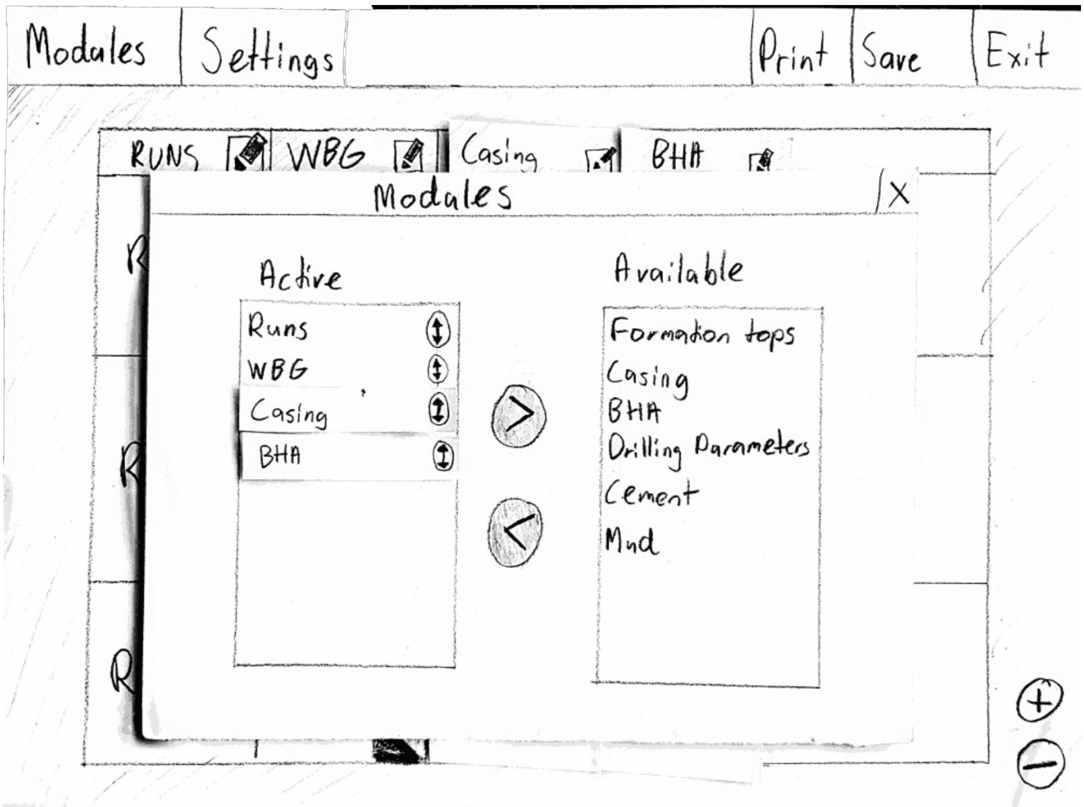
Attributes

Each module contains a set of attributes. These attributes are fields of data that are corresponding to what module they belong in. The user can configure each module with which attributes to display. The menu and interaction method for how to manage the attributes follow two different approaches for the two paper prototypes. The two menus are shown in Figure 17.

The user has to, in the first paper prototype, click the title of the module in order to

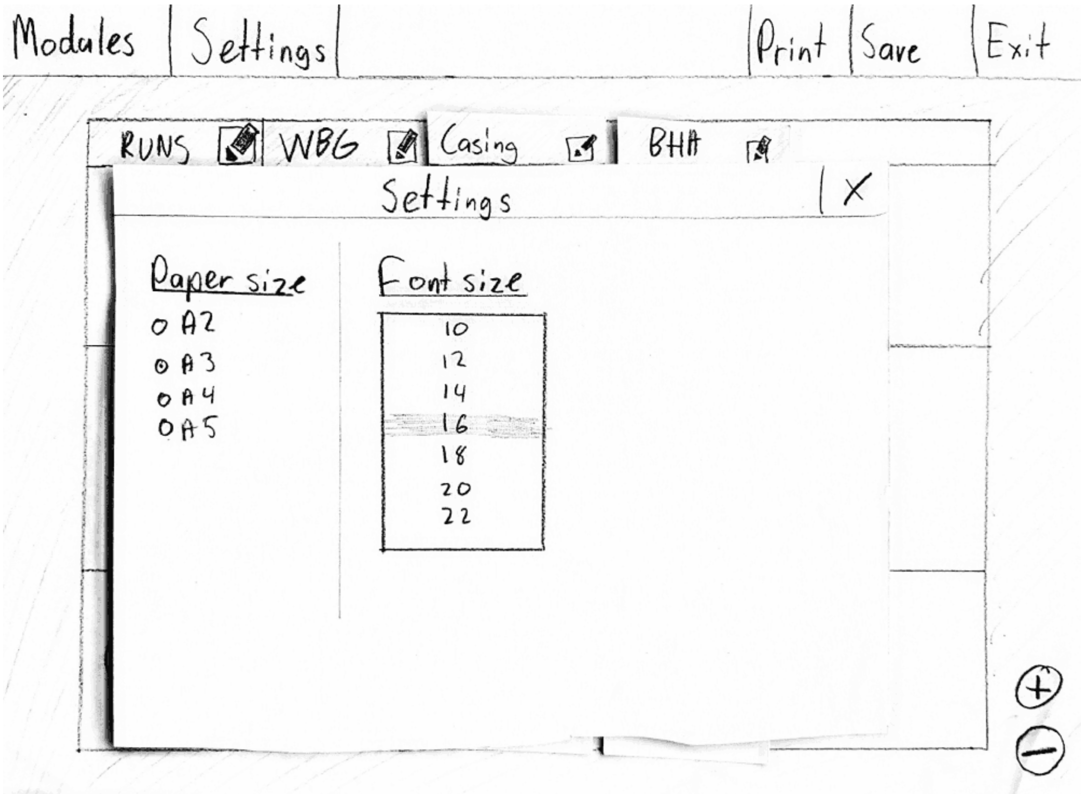


Paper prototype 1



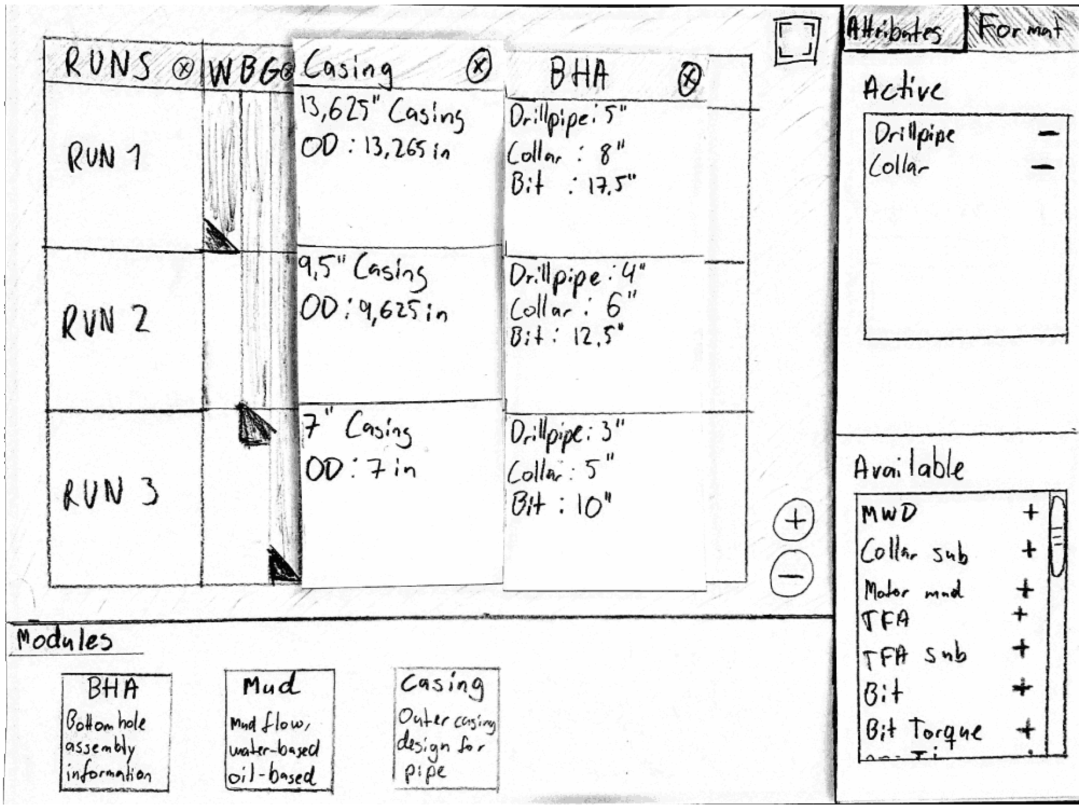
Paper prototype 2

Figure 15: Paper prototype modules screens

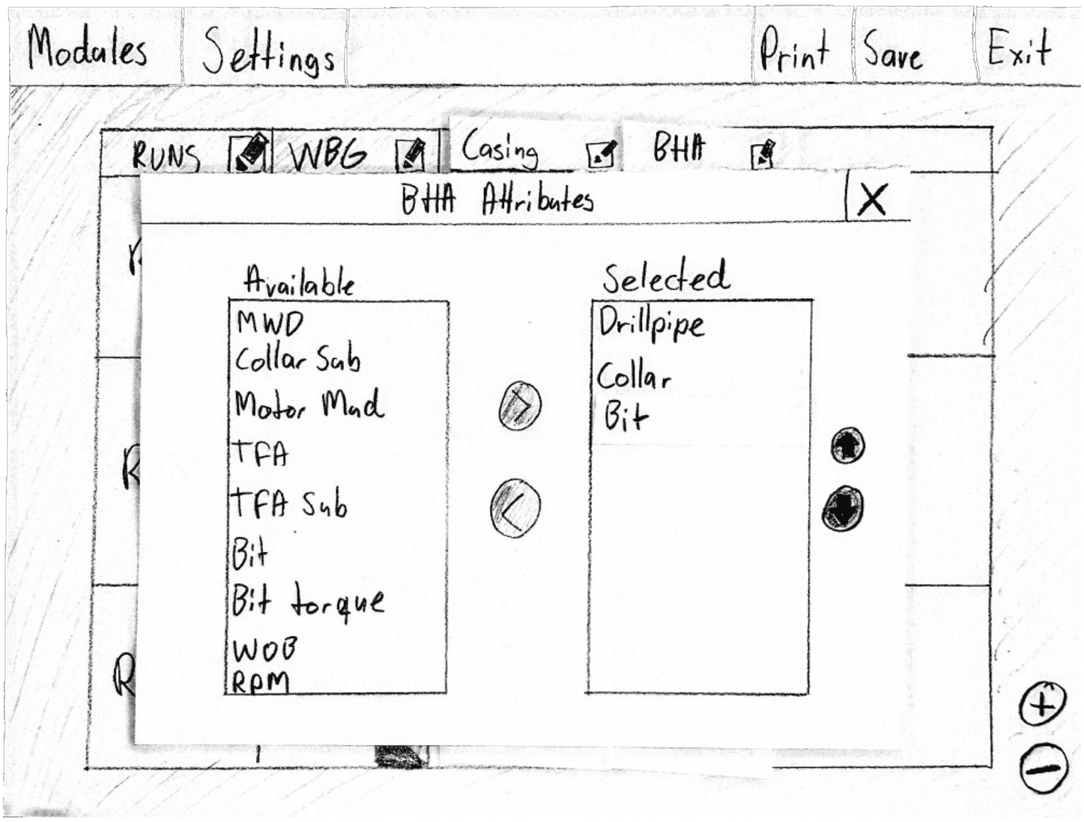


Paper prototype 2

Figure 16: Settings window for the second paper prototype



Paper prototype 1



Paper prototype 2

Figure 17: Attributes menu to configure each module

change module-specific parameters. When clicking the module's title, the whole module gets selected, and the sidebar changes to the attributes menu as shown in Figure 17. The selection of the module was envisioned to be marked visually with a colored border around the module, but this was not implemented due to limitations of the paper prototype method. One can select any module by clicking its title, and by doing so change which attribute menu that is showing. In the side panel, the *"Active"* and *"Available"* attributes are displayed as two separate lists. Adding or removing one of these attributes is done through the plus or minus button on its right. Any click on the blank areas of the canvas deselects any selected module, which removes the attributes panel and returns the general global settings to the side panel.

The attribute panel in paper prototype 1 is also divided into two tabs. The second tab in the attribute menu is called *"Format"* and is intended to control specific formatting for individual modules. This was however not a part of this user test, and the user interface for these options was therefore never drawn.

To change module attributes in the second prototype - you click the edit button that looks like a pencil, located on top of the module on the left of its title. The pencil icon is a common icon for representing editing. A click on this button opens a new window similar to paper prototype 2's modules window. It lists all available attributes in one list, and a second list for what is currently being displayed. The interaction is similar to the modules window, and again, changes here will render continuously behind the window.

6.3 Testing of the Prototype

The testing of the paper prototypes followed the six steps described in Section 4.4. A notable divergence, implied from the fact that this was a test of a paper prototype, meant that the test manager also had to act the role as *"computer"*. A paper prototype has physically loose paper elements that have to be placed on the table as both a feedback and a reaction to the user's input. These pieces of paper have to be manually moved and managed to correctly emulate what a computer would present for the user in a functioning user interface. This is a software developing method referred to as *"Wizard of Oz"* prototyping, and it is often used when testing early user interfaces that lacks the underlying technology *[89] (Snyder **2003** p.92)*.

The prepared tasks and questions, mentioned in the fifth point under Testing Method in Section 4.4, were the following for the paper prototype:

Tasks:

1. Add the modules *"Casing"* and *"BHA"* to your Well Montage

  1.1 Look at the result

2. Add the attribute/field *"Bit"* to your BHA-module

3. Reorder the position of your Casing- and BHA-Module

  3.1 Look at the result

4. Delete the *"Casing"* module

5. Change the font for your Well Montage

6. Zoom in/out your Well Montage

Questions:

A Which version did you prefer? Prototype 1 or 2?

Continuing the list of steps under Testing Method in Section 4.4, the sixth point there allowed for questions and comments from the users. This lead to the question of which paper prototype the user preferred. We asked additional questions otherwise not mentioned for each interviewee as a free-form conversation with regards to how they felt about each prototype and specific features of each. It was done, however, with the comparison of each prototype in mind and to understand their experience more clearly.

The Selection of Users
From Figure 5 on page 27, which describes what users one should test the prototypes on, it is clear that the most applicable user lacks both IT expertise and domain knowledge. This is true especially in this early prototype with limited domain related topics. In fact, the interactions in the paper prototype did not require any knowledge of well planning nor the usage of domain-specific tools. We thereby wanted to find those on the lower end of the matrix in terms of domain knowledge and IT competence. To define having *lacking IT knowledge* are somewhat difficult to define, as the lower end of knowledge of using IT systems can be highly dependent on the user segment one wants to test, and are therefore a relative term. The preliminary research last semester revealed that DrillPlan's target group had good general IT knowledge, meaning they all should be comfortable with using IT devices daily. Especially so with the first-hand users of Well Montage, as these users are part of the technical development. We, therefore, felt it representative to test the initial prototype on interdisciplinary campus students.

6.4 Results and Implications

Comparing the two prototypes there is advantages and disadvantages to both. The intuitive and easy to use drag and drop feature of adding and reordering modules showed to be great for users that are used to similar interactions on e.g. touch devices. It was, however, limited if one has a lot of available modules as the current design does not allow for representing manifolds of modules. The bottom panel of the first paper prototype can remedy this by including a scroll bar, enabling a larger number of modules. This can, however, quickly become overwhelming and difficult to navigate as the design only allows for one row of modules, and because the action of horizontal scrolling is not recommended *[102] (Leavitt, Shneiderman,* et al. ***2006*** *Chapter 8:1)*.

The slower windowed lists of the second paper prototype can prove more useful for large numbers of modules. It showed, however, more cumbersome to use as it demanded more clicks for adding each module. It is also arguably more challenging in terms of usability to represent a canvas of columns as a windowed, vertically organized list. It lacks the correct mapping of vertical and horizontal information. The window also has a negative *"take-over"* effect on the screen. It acts as a pop-up, hiding most of the main information of the canvas that you are trying to change. This is contrary to the standard of *What You See Is What You Get (WYSIWYG).*

There were some issues that were directly affected by the prototype being a paper prototype. Firstly, some users complained it was intuitively hard to differentiate a clickable button from plain text or headings. *"How do I know if I can click this"*, a user said. This is a direct limitation of the paper prototyping method. One loses the animated hover functionality as a visual cue, as well as having a lacking visual representation in terms of color and shadows that usually indicates if a button or other elements are clickable or not.

The second issue that most users were faced with, as a direct consequence of the paper prototype method, was the side menu of paper prototype 1. The right-hand side menu was context sensitive, meaning that the menu was supposed to show information and options relevant to which module the user had selected. Which module the user had selected was however not visually communicated in any way other than a verbal cue. The intention was to have the border of the selected module change color. The selected module was also not explicitly written as a title in the side menu. We thought about implementing both of these parts involving borders and titles but concluded that having this would be difficult to communicate with paper. It was instead explained to the user that elements could be selected and verbally indicated that the module was selected. This gave almost every users problems with regards to going from the context-sensitive menu back to the global menu. Often resulting in the test manager having to give tips that a module was selected. The grayed-out button with the title *"Format"* also mislead some users into thinking that this was the way back to the global settings menu.

There was also some general comments with regards to smaller parts of the user interface. Related the context-sensitive menu in the first paper prototype - a user was looking for a cross to exit the menu. After asking the user to exit out of the context-sensitive menu, they said *"I literally have no idea how to do that, since there is no cross icon to exit it"*. This cross would enable them to close the current attributes menu and regain the settings panel. Users were also missing a cross for deleting each of the modules in the second prototype. The graphically represented alternative for having a cross directly located on the canvas, like in the first paper prototype, was preferred, regardless of which prototype they tested first.

Before starting the second prototype a user commented that *"the top row buttons on the second paper prototype looks like a dropdown menu"*. This user clearly expected the buttons to behave similarly to what native Microsoft and Apple programs do, with a dropdown revealing more options under the given category. Microsoft's Notepad is a perfect example that includes this kind of top row menu, which has been unchanged since its initial release

	T1.1	T1.2	T1.3	T1.4	T1.5	T1.6
Paper proto 1	0	11	2	0	18	0
Paper proto 2	3	1	11	2	1	0

Table 3: Total mistakes the test users did per prototype for each task

in 1985[18]. This may be a problem again with the format of the prototype and its low fidelity. This exact way of displaying menus is currently standard in DrillPlan, except that they are always accompanied by an icon to the left of the button text.

A few users also mistook the zoom-buttons for being the way of adding new modules to the Well Montage, as other user interfaces have similar buttons with a large plus icon to add new features. An example of a program that uses this implementation for adding content is the Google Drive application for mobile devices. Here there is a circular blue or red button (depending on the operating system) with a white plus symbol located at the lower right corner. The button is used for adding new files to your Google Drive. The buttons in our prototype were located at the same place and having it being an unlabeled button with only icons to symbolize their purpose, there are reasons for usability questions to be asked. *‘‘It would be easier to understand if the zoom icons also included a magnifying glass, so I know it is used for zooming”*, one user commented. That being said, Google itself has the exact same button layout as our prototype for zooming on their PDF reader on Google Chrome; light circular buttons with black plus and minus symbols. The only difference is that they include a full-screen button right above the zoom-buttons.

Google Drive

Google Chrome

Preferred Version

From Table 3 it is clear that paper prototype 1 is the prototype with the most registered user mistakes. Mistakes are not the same as identified prototype errors - it is rather a metric showing how many faulty operations the user had to undergo in order to complete the given task. In the cases where we had to give hints as to how to move forth in any given task, we automatically added three mistakes to the total of that task. Paper prototype 1 has the most registered mistakes, and also needed the most hints. That being said, from the conversations with the users after completing the tasks for each paper prototype, most users preferred this prototype. One said that *“it is easier and much more intuitive to drag and drop modules on the Well Montage. Using the lists of paper prototype 2 feels old-school and slow”*. Prototype 1 had more user mistakes, but it might be because its more modern approach was less suitable to test properly with a paper prototype. And despite the mistakes, users might have seen its potential, and thereby liked it better than paper prototype 2.

There was, however, users who preferred paper prototype 2. This was said to be because

[18]https://en.wikipedia.org/wiki/Microsoft_Notepad

of the familiarity of adding and removing modules and attributes, which may be similar to programs they frequently use. This bias, in addition to prototype 1's difficulties of reaching through in a paper prototyping environment, may be why these users felt more attracted to the second prototype. Their strong feelings as to why the familiar and *"old-schoolness"* was better than the new and more interactive solution should not go unnoticed. At the end of each of these user-tests, we explained the intention behind how the first paper prototype was supposed to behave. After this, most, if not all, liked the first prototype better than before. The users that preferred the second prototype had a clearly less negative attitude towards the other prototype, compared to their opposites.

System Usability Scale

After the tests, each user filled out the SUS-questionnaire, as seen in Table 4. With an average score of *87.5* across seven tests, one can conclude that the paper prototype has high usability, but not excellent as that would need it to be above 90 *[103] (Bangor, Kortum, and Miller **2008**)*. One can see from the table, two questions have a higher variance than the rest; question 8 and particularly question 1. For question 8, virtually no users understood the meaning behind the word *"cumbersome"*. We had to translate this word for them, which may have impacted the way the question was scored. This is a known problem with the use of the SUS-questionnaire on non-native English speakers *[104] (Finstad **2006**)*. Regarding question 1 there is a clearer reason as to why it has such a high variance. For this questionnaire, we gave the users free reigns as to how to interpret the questions. Question 1 asks them if they would *"use this system frequently"*, and as none of the users were targeted end-users of the system, this got interpreted differently. The answers varied greatly since none of the users actually have any reason to use the application, and therefore no interest in using it. Some user interpreted the question as if they worked in an oil company, and others did not, which explains its high variance. Higher variance can more easily occur with this few test results, so the total score should be taken with a grain of salt.

Free reigns might have impacted the result both positively and negatively. The purpose of doing the questionnaire this way was to make the answers as honest as possible. We did this with a plan for later sorting out the question that had frequently been up for discussion and as a result had ambivalent answers. The free reigns might have helped in questions directly regarding the usability and user experience, but as seen from these results it also gave negative impacts on other questions. For the next iterations, it would be wise to have clearer guidelines for the users as to how to fill out the questionnaire, thereby reducing uncertainties with regards to the users' intent. When it comes to question 1 specifically, it might have to be discarded completely if the tester is not from the targeted audience for the system. There is also a possibility of using a questionnaire that is translated to the user's native language to avoid further misunderstandings that might come from a language barrier.

Next Iteration

Whether to choose one or the other prototype is not straight forward. If half of the user base does not like the method of interaction, it could in the worst case result in a halving

	Q.1	Q.2	Q.3	Q.4	Q.5	Q.6	Q.7	Q.8	Q.9	Q.10	SUS
User 1	5	1	5	1	5	1	5	1	5	1	100.0
User 2	5	1	4	2	5	2	5	1	5	1	92.5
User 3	5	3	4	3	5	1	5	2	4	1	82.5
User 4	1	1	4	1	4	2	5	4	4	3	67.5
User 5	3	1	5	1	5	1	4	1	5	1	92.5
User 6	5	1	5	1	5	1	5	1	4	1	97.5
User 7	1	2	4	1	4	1	5	1	4	1	80.0
Variance	3.62	0.62	0.29	0.62	0.24	0.24	0.14	1.29	0.29	0.57	87.5

Table 4: SUS-table for the paper prototype

of the user-base. There were several users that acted directly negative to the second paper prototype, calling it *"old-school"*, but no one was acting as negative towards the first prototype. As one of the goals of the configurable Well Montage is to encourage the use of its new features and that it fits well into the DrillPlan platform, there is a need for it to be modern. There is, however, no need to explicitly choose *one* of the approaches. The final solution would benefit from being a mix of both approaches, maybe with the possibility of both drag-and-drop, and a separate menu for doing other more data-heavy tasks for practical visualization reasons.

Smaller elements like having both a cross and an edit button on each module is a possible change for the next iteration. So is having a button for full-screen together with the zoom options in the corner of the canvas. If the top menu buttons are included, they should be accompanied with relevant icons to their functionality. Drag-and-Drop interaction should also be continued in the next iteration, but context relevant contents should be less crucial for doing edits. It is also important to consider the number of options that are to be fitted in the menus so that the format of displaying these are not hindering the usability. In the case of most content heavy operation, namely the applying attributes to modules, there is a need for a clever implementation that is both space effective, and that visualizes the content in a meaningful way.

7 Non-functional Prototype

This section describes the second iteration of the project; the non-functional prototype. With the paper prototype tested and evaluated, it was time to upgrade both in terms of scope and fidelity, to implement more of the requirements from Appendix D. The paper prototype was limited in scope by only including some of the core functionality, namely the editor, and not the entire flow of the system. Paper prototyping as a technique is limited with its low-fidelity, so to avoid false problems induced by users positively deciphering the prototype and problems related to paper prototypes slowing down the actions, the next logical step was a digital non-functional prototype *[89] (Snyder* ***2003*** *Page 302).*

Jake Knapp writes the following on the importance of creating an illusion of an actual environment around the prototype: *"(...) you'll make a prototype that appears real, just like that Old West facade. (...) your customers — like a movie audience — will forget their surroundings and just react." [76] (Knapp, Zeratsky, and Kowitz* ***2016*** *p.161).* He continues by describing that the prototype should *"deceive"* the test subject to think it is and works like the real thing by reaching for *"just enough"* by making a *"temporary simulation"* as Knapp describes it. Thereby creating something real enough to evoke the crucial and constructive feedback from the users that is necessary when going forwards.

With the paper prototype, the concept of the editor was tested with several design possibilities as to how to interact with a configurable Well Montage. The problems that arose in these tests were taken into account when designing this prototype. By only including the best initiatives established from the evaluation, and discarding those that did not work, the non-functional prototype should be a significant upgrade in quality. In addition, some initiatives were re-evaluated after the upgrade from paper to a digital environment to rule out method dependent problems. Feedback as to what buttons are clickable and other context-sensitive information are more easily communicated on a non-static, digital prototype. The scope of the non-functional prototype was also extended beyond the selected core concepts of the editor, by including all the aforementioned interactions in the map of Figure 12. The *Well Montage Dashboard* that covers the point of *"Create new Well Montage"*, which functions as a starting position for the prototype, gives the appropriate initial context to the user. Thereby, enhancing the facade by including what Knapp describes as an *"opening scene"* from which the prototype should take place *[76] (Knapp, Zeratsky, and Kowitz* ***2016****).*

Following Figure 5 we wanted to test a larger part of the user-test matrix by assessing interactions by users with greater domain knowledge. The non-functional prototype was therefore tested on Schlumberger employees. They have a domain knowledge above average, are comfortable within the DrillPlan framework, and can give more specific feedback regarding this.

7.1 Prototyping Software

Moving from paper to a digital prototype renders a need for a prototyping tool. The tool has to be able to offer sufficient and necessary functionality to cover the needs when creating a non-functional prototype. It should be simple and quick enough so that the workflow is not too cumbersome, but not too simple as to limit the prototype scope in any meaningful way. To help in this decision we created a comparison of a selection of well-known prototyping tools. This comparison can be seen in Table 5.

	Figma	**Marvel**	**Axure**	**InVision Cloud**
Live Edit	Yes	Yes	Partially	Yes
Platform	Browser & Windows	Browser	Only Windows	Browser
Commenting	Yes	Yes	Yes	Yes
Components	Yes	Yes	Yes	No
Price	Free*	Free*	30-days Trial	Free

* Free, but with paid plans for additional features

Table 5: Comparison of prototyping tools.

From the table there are five aspects that were important for our prototype tool of choice:

1. *Live edit* - enables us as a team to collaboratively work on the prototype at the same time. This maximizes the output rate and limits problems with merging.
2. *Platform* - limitations on our faculty machines with administrator accessibility as well as local machines running Linux, made the platform availability a factor.
3. *Commenting* - in order to have a responsive workflow where both team members can contribute by asking questions or directly suggest edits on different elements. Having the feature embedded in the tool is, through experience, almost to regard as a necessity to ensure efficiency and minimize communication issues.
4. *Components* - the adding and removal of interactive components made inside the program, and the reusability and customizability of said components.
5. *Price* - less important, but it is preferable to work with something that is affordable. It makes the choice of changing program easier to do on a later stage if it renders necessary. It makes it more appealing for subsequent projects, and it limits the complications and delays with applying for funds.

With these aspects in mind, it would seem that either Figma or Marvel would be an adequate choice. However, Marvel has, by previous experience of using it, showed to be too simple for its own good. It quickly becomes challenging to manage prototypes with many graphical elements when dealing with multiple referencing screens, as it lacks inheritance of components. The tool has also been prone to synchronization issues with multiple users simultaneously editing the same prototype. In our experience, it thereby limits the workflow and would make a less than optimal tool of choice. Every other prototyping tool in the table was also tested and explored to see how the workflow was, and if one could potentially disregard some of the aforementioned aspects. From review recommendations and through testing with the five aspects in mind, the decision came down to Figma. It checked all the boxes and featured a more varied set of tools than Marvel with better management options and without the performance issues.

As the prototype grew in size, so did the complexity inside Figma, as seen in Figure 18. It might look messy and hard to comprehend, but with a bit of practice, it became very natural to use, despite its apparent chaotic nature with lines going everywhere.

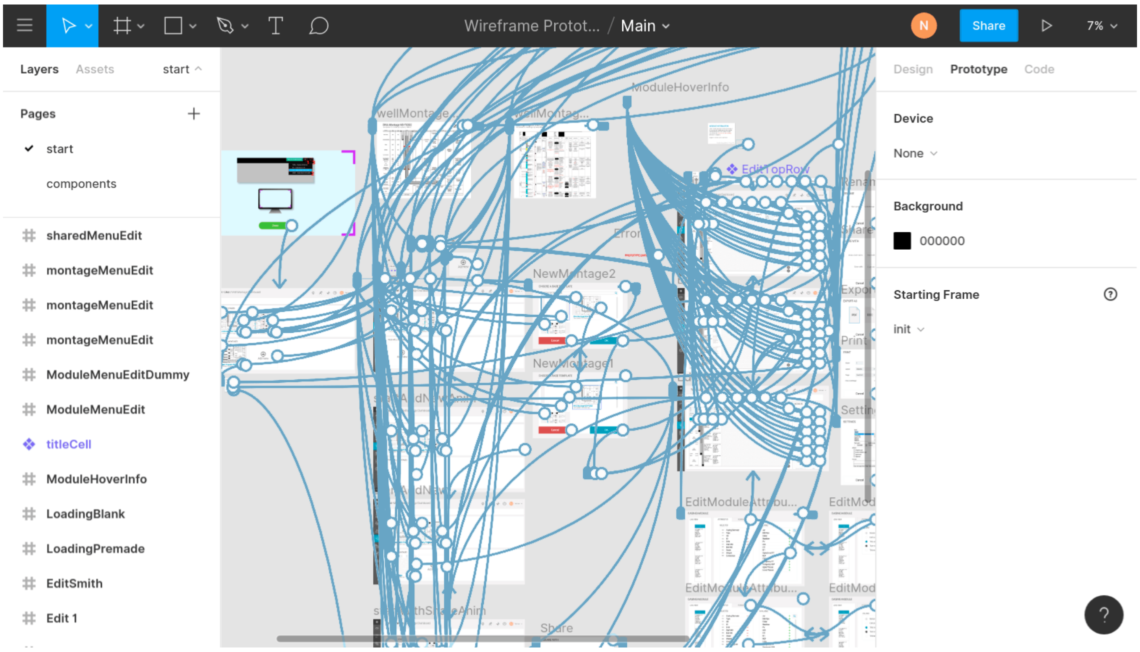


Figure 18: The figure shows how the workspace of Figma looked after the completion of the non-functional prototype. The blue lines represents all the possible actions the user can complete between the screens in the user-tests.
The color of these lines have been darkened slightly to improve readability.

7.2 Prototype Description

For the prototype, to enhance the aforementioned facade, we decided to set it inside a shell that looks like DrillPlan. Meaning, having a non-functioning navigation sidebar and top-bar in all parts of the prototype that is similar in appearance to what is present in DrillPlan. In addition, without following any *"official"* guidelines, the prototype is following the general design choices already established in DrillPlan, particularly with regards to color choices, buttons, and icons. Thereby making it easier for the test-users to envision the prototype as a part of DrillPlan, and not a separate application detached from the workflow.

Main Menu

The first screen in the prototype, seen in Figure 19, is the start page of the application. This is where the user manages their Well Montages, thus the name *"Well Montage Dashboard"*. It contains Well Montages made by the user, as well as Well Montages shared by others. From the dashboard, users can by clicking the three horizontal dots located on the bottom right corner of each module container; remove, rename, print, and share their own Well Montages. For the Well Montages shared by others they can by clicking the same button; view and edit them, view information about who shared and updated it last, and print them. The sidebar on the far left was a simple screenshot taken from DrillPlan, with the blue button displaying that the user is currently in the *"Plan and Validation"* section of DrillPlan, where Well Montage is located as of today. This and the top-bar with the bread crumb menu, a few buttons, and user information, is strictly non-functioning and thereby not interactive in any way.

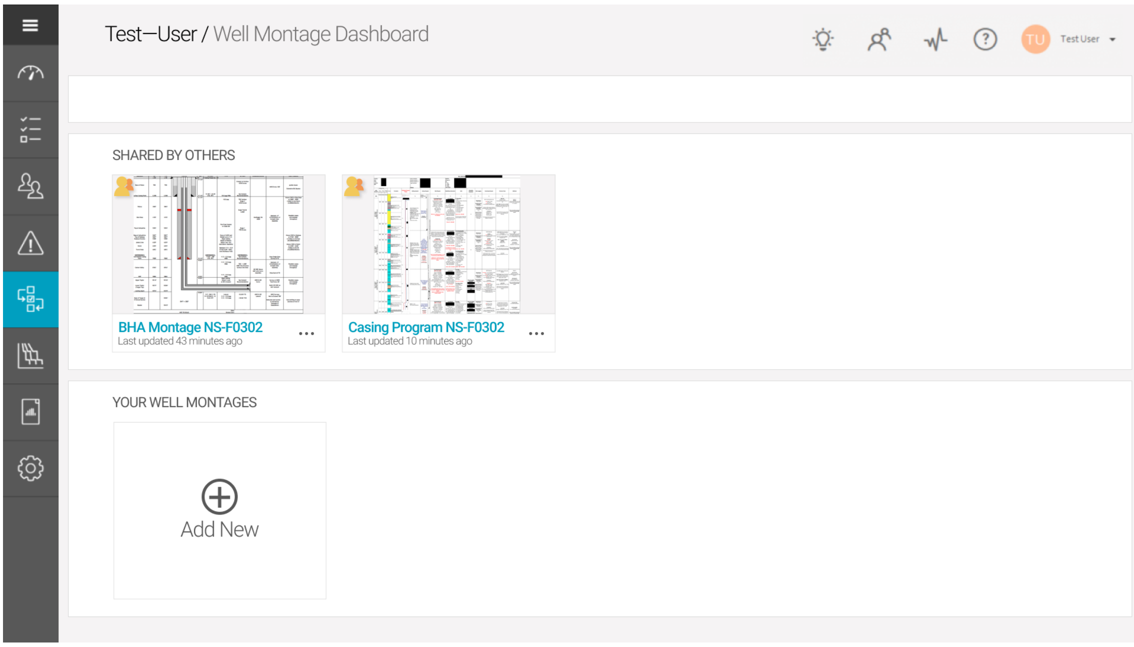


Figure 19: Well Montage Dashboard of the non-functional prototype

When clicking the *"Add New"* button, a window appears where the user can choose between predefined templates, or basing them on other preexisting shared Well Montages. In the prototype, only a *"Basic"* preset template is present, to avoid clutter, but additional templates were envisioned. This was a simplified version of what the final product would look like, where there would be a clearer separation between those that are predefined presets, and those that are based on already made Well Montages. It was done to avoid the additional work of introducing concepts of the application which are not the main point of interest for this prototype.

Editor

The editor-view is a continuation and combination of what was made during the previous iteration in the paper prototype, and can be seen in Figure 22. It contains the editable

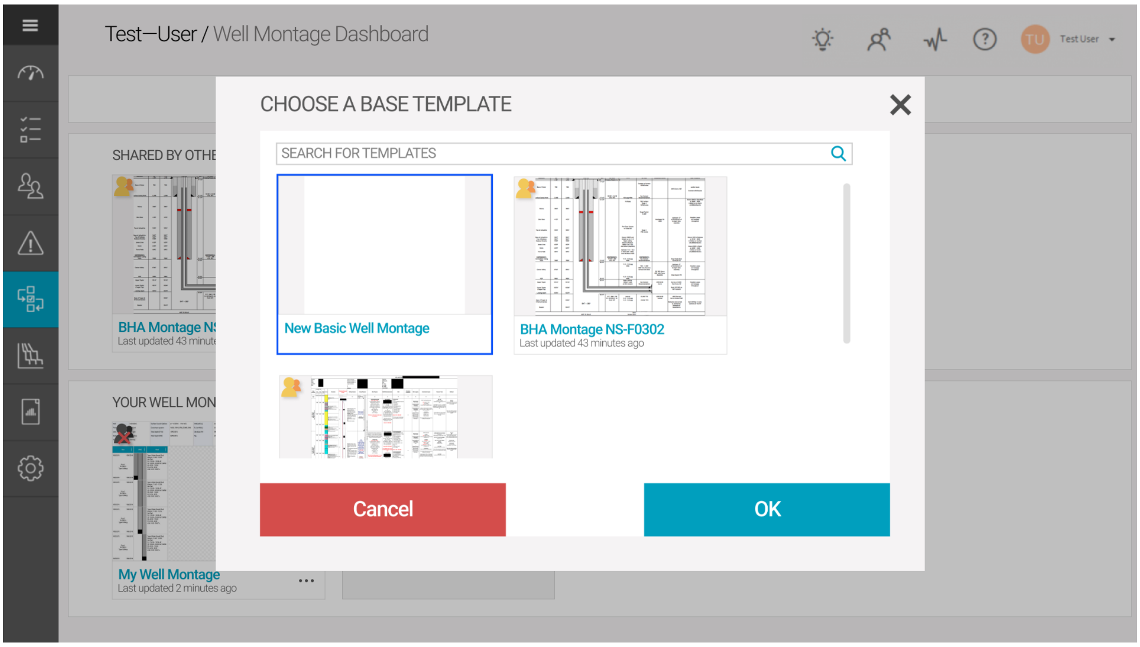


Figure 20: Creating a new Well Montage in the non-functional prototype enables the use of templates

canvas with additional buttons on the top of the screen, now with icons in accordance with DrillPlan's design. On the right side is a menu containing the addable modules. As most users preferred the drag and drop of modules, this was decided upon being the main way to place and move modules. The module menu is, however, moved away from the bottom of the screen for practical reasons and to enable more modules to be displayed at any time. A menu on the bottom of the screen, as seen in the two illustrations marked "(1)" in Figure 21, demands too much of the screen real estate for a canvas of a strict A4 relative size. It would also force horizontal scrolling to encompass more modules. With the new layout, one saw an increase in the actual displayed canvas size of up to 25% for the most used aspect ratio for monitors and screens of any kind. It also gave the possibility to have more modules viewable at the same time. Screen ratio in between 4:3 and 16:9 was chosen to be supported as it would according to StatCounter[19] account for a minimum of 70.9% of all worldwide users displays between April 2018 and April 2019. With only 4.9% guaranteed outside this ratio.

Information and dependencies of modules can be shown while hovering the mouse over each specific module in the side menu. The prototype shows the same text for each module, due to limitations with Figma, but a fully fledged version would have context relevant information dependant on what module the user hover over. This feature is shown in the right of Figure 22 where the user is hovering the *"Offset"* module and viewing more information.

The heading for the *"Module Menu"* panel on the right side is designed with the ability in mind for it to enable future expansion of the panel into tabs. It included only a *"Module"*

[19] `http://gs.statcounter.com/screen-resolution-stats`

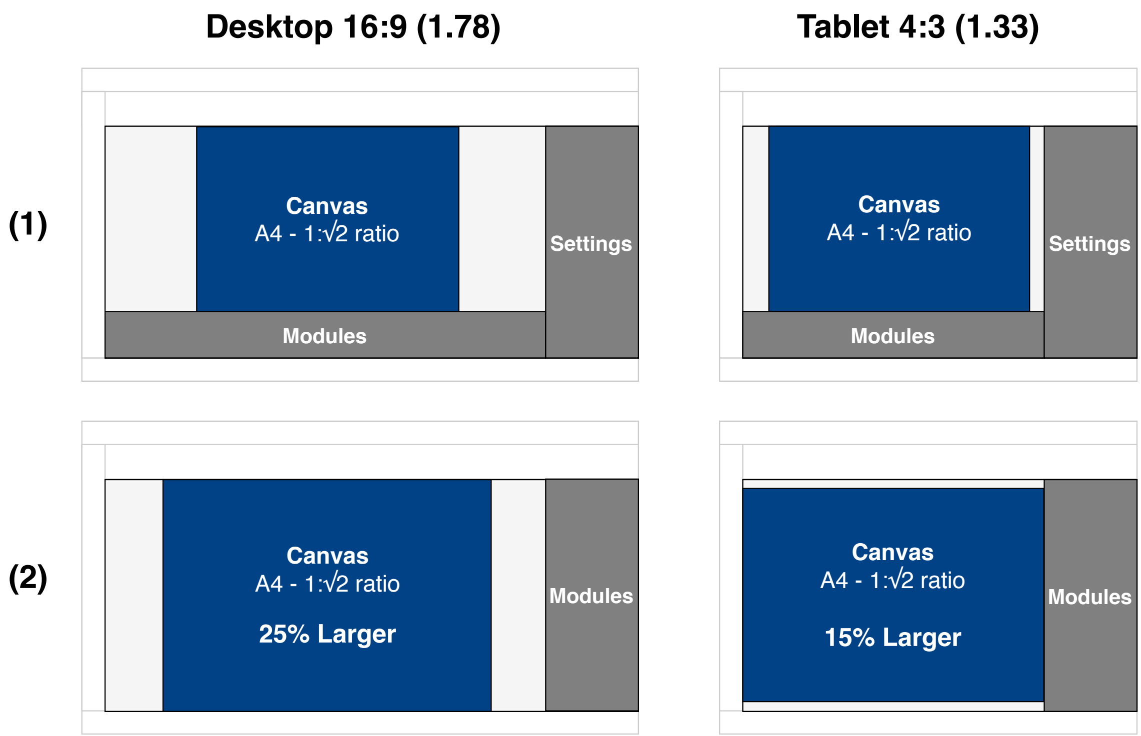


Figure 21: Screen ratio for the canvas in the editor of the non-functional prototype

tab in this prototype, but a later expanded version of the program could require the addition of new features that utilize this extra space for tabs. When comparing the design of the first paper prototype one notices that the non-functional design has only one extra properties panel in addition to the canvas. A tabs system would, therefore, increase this number to at least two like in the paper prototype.

The inclusion of a context-sensitive menu was something that was tested in the paper prototype. It explicitly tested the concept of selecting the module you want to edit by clicking on its title, and then let the program show you information and options related to your selection. This interaction did not seemingly communicate well, at least not in the paper prototype. This may, as discussed in Section 6, be due to the limitations of the testing method itself, and it was therefore originally intended to quality check this claim as part of the non-functional prototype. However, it was dropped due to limitations in the prototyping tool, making the functionality too time-consuming to mimic. The concept of context-sensitive menus was therefore at this point put on hold.

In the far bottom right of the canvas, we included an updated version of the zoom buttons from the last iteration. As talked about, it now features the inclusion of a full-screen button to facilitate the perception that the plus and minus buttons function as zoom-buttons.

Customizing Modules

The interaction for the customizing of modules was, like the editor, also drawn from experience with the paper prototype. The menu is available for each of the modules placed on the canvas by clicking the three vertical dots to the right of the module's title. This

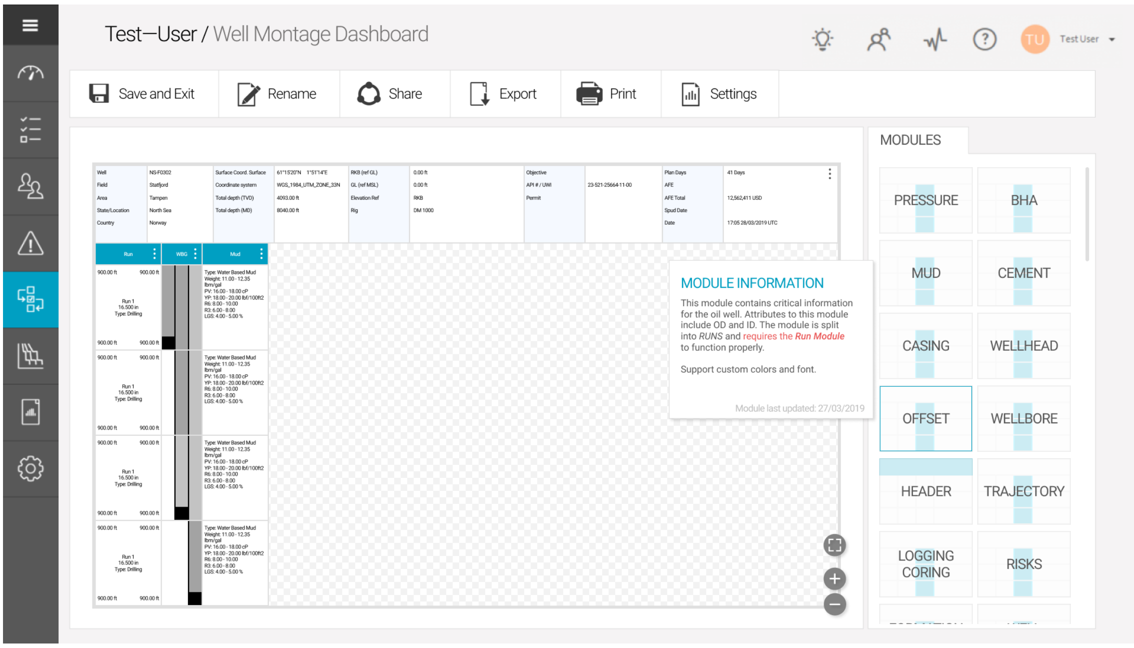


Figure 22: The editor in the non-functional prototype

button would open a dropdown menu where the user could choose between *"Edit"* and *"Delete"*. The edit buttons open a pop-up window inspired from paper prototype 2, as this seemed to be the best way of representing all the possible edits without using the disputed technique of context-sensitive menus. It is arguably the best solution for utilizing the space available. A screenshot of how this menu looked can be seen in Figure 23. A test-user in the previous iteration pointed out a problem with this particular design compared to the context-sensitive side menu: *"A pop-up window like this, is covering almost the whole screen, and is blocking the view of the workspace"*. Edits done in this pop-up window would hinder the user in knowing how these edits actually affected the look of the Well Montage. The pop-up window can accommodate this problem in one of two ways. You could make the window semi-transparent and have the canvas in the background update continuously while the user edits. Or, one could let the window share enough information so that the user fully understands the implications of the edits. A window with as many options that we wanted to include, would severely clutter the view if it were to be semi-transparent. The solution, therefore, calls for giving the user a fuller picture. We decided that the best way to do this was an addition of a live preview of the module that is being edited. It shows the edits as they are being made next to the options so that the user knows exactly what the output will look like. This approach is similar to how the context-sensitive menu of the first paper prototype handles this matter, but it renders out the problems of having to select an active module for editing without the proper feedback or motivation. Examples of this way of interaction can be seen in many of Adobe Photoshop's different filters. Here you often get a pop-up window with options next to a preview of the actual result.

The way of editing the specific attributes making up the modules was heavily inspired by paper prototype 1. From the first paper prototype, we had two lists containing the available attributes and the selected attributes, i.e. the attributes that are viewed. The idea was to let the users easily move attributes from one list to the other. Thus, choosing which attributes that the modules should be displayed or not. It continued the drag and drop functionality for reordering the attributes introduced in the first paper prototype. The lists in the non-functional prototype only differ from the first paper prototype in that they are two vertical lists placed next to each other, rather than two vertical lists stacked on top of each other. This change came naturally with the increased screen real estate of having a widescreen pop-up window instead of a thinner side-menu. It also made it possible to support an increased number of viewable list elements at once. One could argue that it was easier conceptually to have a scroll bar next to both lists and have them scroll independently, compared to having two scrollable lists on top of each other like in the paper prototype 1. In addition, the user-test from the paper prototype revealed that having two lists side by side had a higher sense of familiarity for several of the test-users.

The attributes window contains a second tab called format. This tab enables the user to change visuals for each module. It seemingly allowed for editing the colors, font type, font size, and text float. In reality, only the float to center was implemented for user-testing, the rest was non-functioning. The designed window for this iteration can be seen in Figure 24

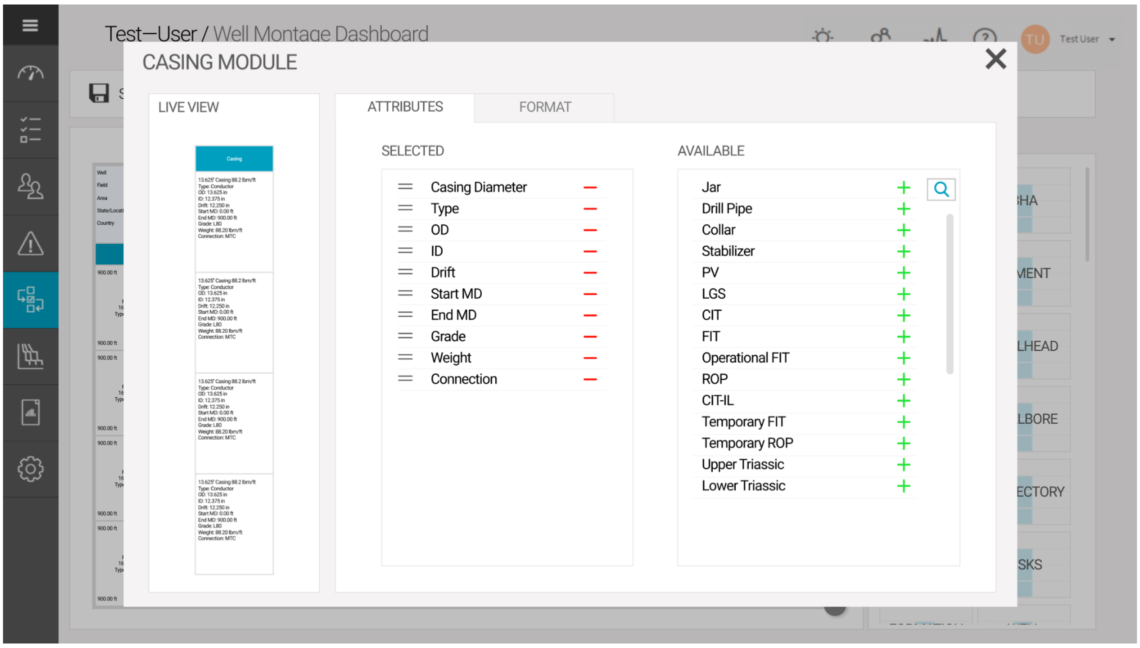


Figure 23: Edit which of the attributes of a module is currently selected

7.3 Testing of the Prototype

The testing of the non-functional prototype was conducted mostly on par with how the paper prototype was carried out. Documents for each of the non-functional prototype tests consisted of: an *Information Letter*, a *Test Information* for the prototype, an *Interview*

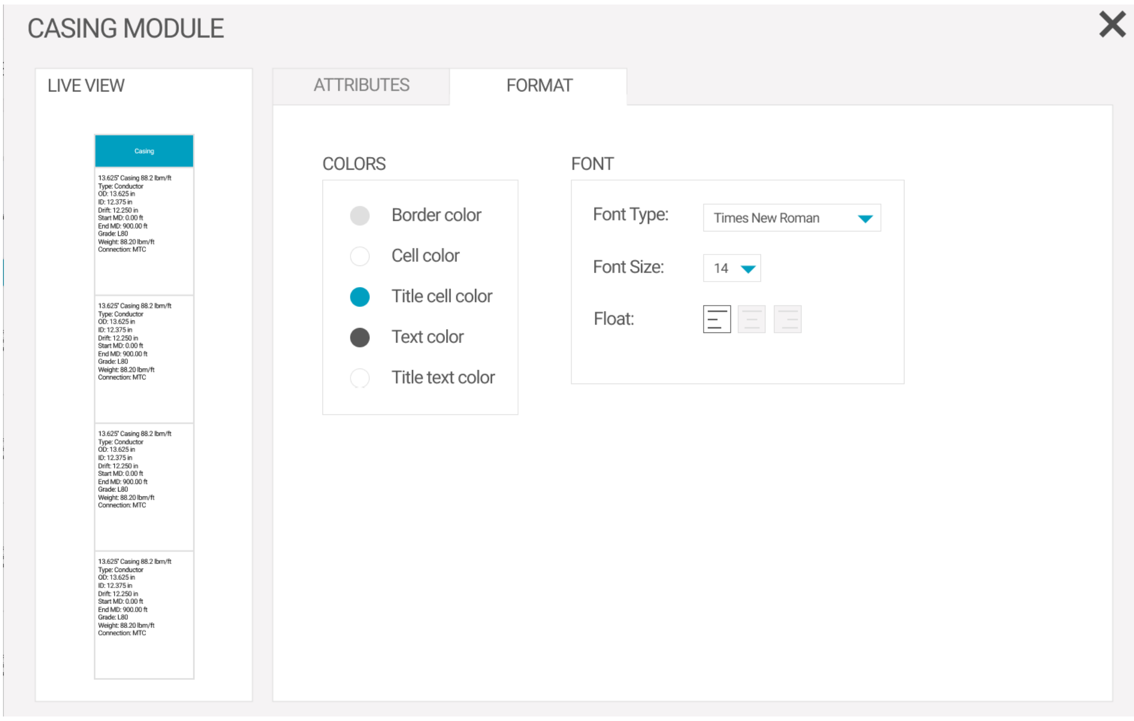


Figure 24: Formatting options of individual modules in the non-functional prototype

Guide, and a *SUS questionnaire*. The *Information Letter*, identical to the one in the paper prototype, presents the project as a whole with legal information regarding the handling of the test users' personal data. The *Test Information* describes how the test will be conducted and its context. The *Interview Guide* is the questions and tasks that are tested in the user-test. And finally the *SUS questionnaire* was identical as in the last iteration.

The non-functional prototype differed in having a larger scope and more functionalities up for testing. This increased the complexity and the number of tasks each test-user had to do. The user-tests was, like with the paper prototype, initialized with two pilot tests. The increased number of tasks, as well as the increased complexity, meant more problems were discovered in these pilot-tests, both with regards to task orders, but also regarding specific phrasing of each question. The pilot-tests also revealed minor design faults in the prototype. After the completion of the two pilot-tests, these issues were addressed and fixed before the tests with the Schlumberger employees. The final version of the tasks for the prototype was:

1) Which Well Montages are currently being shared by others?

2) View the *BHA Montage NS-F0302*

3) Who shared it?

4) Remove your Well Montage.

5) Add a new Well Montage based on the *BHA Montage NS-F0302*

6) Zoom in on the Well Montage

7) Download it in a PDF format

8) Save and Exit

9) Remove your newly created Well Montage

10) Add a new Well Montage of the type Basic
This prototype includes 4 modules in the basic template layout.

11) Add a Casing module to your Well Montage

12) You would like another font on your Well Montage - change it.

13) Save and Exit

14) Rename *"My Well Montage"*

15) Remove the Casing module from *"My Well Montage"*

16) What module does the Cement module require to function properly?

17) Add a Casing module again

18) Center the text on the Casing module.

(a) Revert the changes

19) Add *Operational Fit* attribute to the *Casing module*

20) Rename the Well Montage again

21) Now print the *Casing Program NS-F0302*

22) Who shared the *BHA Montage NS-F0302*

23) Share *"My Well Montage"* with Everyone within company (must be on start screen)

(a) How can you tell it is shared?

24) Stop sharing *"My Well Montage"*

(a) How can you tell it is not shared?

After all the tasks were done, some followup questions were asked:

A) Adding modules; Click versus Drag and Drop? Why?

B) How would you reposition your modules on the Well Montage?

C) How would you reorder the attributes for each module?

D) Did a live preview of the editing of modules give you the necessary insight?

(a) Did you miss a save button?

E) Shared by others Well Montages vs Your Well Montages? Explain your thoughts on these two on the main menu.

F) Explain how you think sharing of Well Montages work now? (mental model)

(a) Additionally, how do you think they should work?

7.4 Results and Implications

After testing the prototype on the five selected Schlumberger employees, the general consensus was that the prototype was very functional, intuitive and a nice preview on a new and needed feature in DrillPlan. As mentioned, despite us not following the *"official"* DrillPlan UI guidelines, many test-users said that it felt like a part of DrillPlan, and nothing stood out as too out of the ordinary. *"This looks just like DrillPlan"*, a user encouraged just as the testing began. Thereby achieving the facade by having a veil of realism to the prototype.

One user tried to click the side-menu as a way to get *"back"* to the main menu of our application, as this is regularly a flow present in DrillPlan. The only way to get back to the main menu was through the *"Save and Exit"* button. Some users found this confusing and commented: *"What if I only want to save, but not exit the editor"*. This feature was, however, not implemented as it was too hard to mimic in a non-functional prototype. This was the only negative aspect of using the DrillPlan facade. Most users commented on how they liked that it followed the general design principles of DrillPlan, and unfamiliar design principles were not too foreign than that they were promptly understood. This may be due to our use of familiar design principles that the users easily recognized from other applications. Example of which can be seen in Appendix E.

The number of overall mistakes made by the user was also significantly lower than on the paper prototypes. Despite the added complexity of a larger scope and more tasks, the users managed to find their way around to the correct functionality in the program. There could be several factors as to the reasons for this decrease in mistakes. It would be unjustified to assume this decrease was only due to a good design, though a bad design would certainly induce a lot of problems. Firstly, the decrease in overall mistakes can be explained by the upgrade from paper to a digital prototype. It helps in communicating the features through more proper feedback, clearer boundaries, and improved visuals. Thus decreasing false errors that are often present in paper prototypes *[89](Snyder **2003**)*. It is also probable that the new medium helped with the immersion for this test's more domain experienced users, compared to the paper prototype tests where the users were strangers to the system. Secondly, one could argue that the increased particularity of the tasks given in this test, guided the user too much towards the right action. There is no definitive answer as to how to make the test guide, but we wanted a balance between having full test coverage of most of the functionality, in addition to having some tasks open to interpretation. More open tasks could have lead to more user errors and a lower definitive cover of the individual elements of the prototype. It would produce more variation in the documented result, which would be harder and less clear to compare and conclude empirically.

Out of all the tasks given to the test-users, task 3 regarding who shared a Well Montage, caused the most user errors. The intended action was to hover over the orange *"shared by"* icon located on the top right corner of the corresponding Well Montage, as seen in Figure 25. However, as the previous task was viewing the same exact Well Montage, there was a misunderstanding as to what *"Last updated"* entailed. The intention was that the

Well Montage was last updated by Nancy Williams, but shared by J. Smith. So the blurred lines between the definition of *"shared by"* and *"Last Updated"* caused the users to answer incorrectly. In addition, the information that was crucial for the task was hidden behind the action of hovering over an apparently insignificant icon. It is thereby not in line with the principle of affordance, as the affordance is not available until an action has taken place *[91] (Norman **2013**)*. This is also called *"hidden affordance"*.

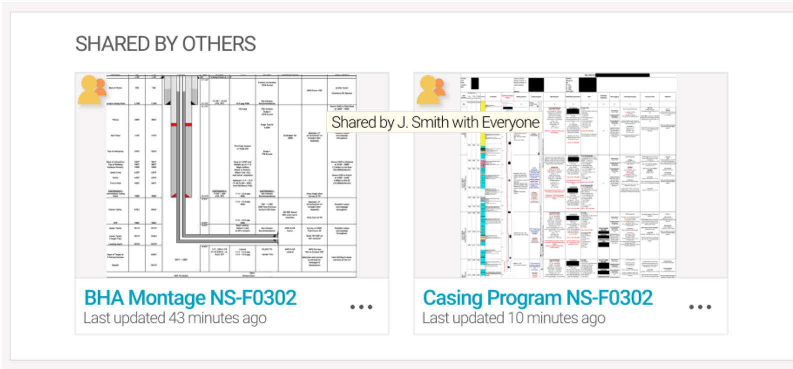


Figure 25: See who shared a Well Montage, by hovering over the orange icon

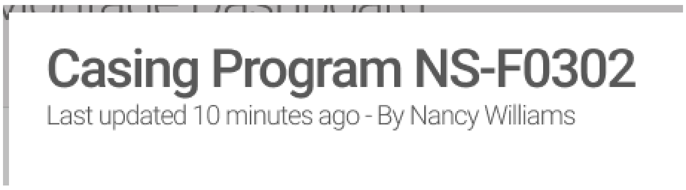


Figure 26: The ambiguity of *"Last updated by Nancy Williams"* in the preview of the Well Montage in the non-functional prototype

There were some problems regarding the creation of a new Well Montage based on *"BHA Montage NS-F0302"*, namely, task 5. Some thought this meant navigating to the listed Well Montage under shared Well Montages, then clicking the three dots opening the dropdown menu where a copy button was to be located. All these users that tried this approach quickly understood where to look next, and continued by doing the intended action. One other user thought that one had to open the editor for the appropriate Well Montage, and then copy its content from there or have a dedicated menu for this feature inside the editor. In reality, we wanted the users to add a new Well Montage, and in the pop-up window choose one of the existing Well Montages as a template for the new one. This communicated well enough based on the feedback given when the users got the pop-up window. Adding a copy directly in each Well Montage's drop-down menu is a valid addition to encompass most of the user preferences.

There was also some confusion as to what it entailed that a Well Montage showed up in the template browser after the user clicked to add a new Well Montage. The pop-up window showed Well Montages that already existed on the first screen. *"Are they their own thing, or are they somehow related to the shared by other montages from earlier?"*, asked one test-user. When creating new Well Montages, it should be clear as to what

	Q.1	Q.2	Q.3	Q.4	Q.5	Q.6	Q.7	Q.8	Q.9	Q.10	SUS
Test-user 1	4	1	5	1	5	1	5	1	5	1	97.50
Test-user 2	5	1	4	1	5	2	5	1	5	1	95.00
Test-user 3	4	1	4	2	4	3	4	2	4	2	75.00
Test-user 4	4	1	5	1	4	3	5	1	4	3	87.50
Test-user 5	5	1	5	1	4	1	5	1	4	1	95.00
Test-user 6	4	1	5	1	4	1	4	1	4	1	90.00
Test-user 7	3	2	5	2	4	2	5	1	5	3	80.00
Variance	0.48	0.14	0.24	0.24	0.24	0.81	0.24	0.14	0.29	0.62	88.57

Table 6: SUS-table for non-functional prototype

are predefined preset-templates that are made in advance, and what is based on already existing and shared Well Montages. In the final version, there will be necessary to have a clearer description and distinction for the Well Montages that can be copied. The notion of calling a copyable Well Montage for a template is somewhat imprecise and should be clearly defined.

The header module was pre-added to the so-called blank Well Montage template. This was partly as a reference as to how the current Well Montage looks, but also to make the test users more familiar with the layout of a partially empty Well Montage. A test-user mentioned the possibility of *"clicking the three dots menu in the top right to access global Well Montage settings"*. This button was supposed to be a local button for editing the header module, but can clearly be misunderstood for a global settings button. Which begs the question; how can the header be implemented so that it is still an editable module, and this is made clear to the user? Simultaneously while containing unique functionality since it does not rely on runs such as the other modules do. It would require its own attribute menu which would function differently than the rest, since it potentially could have more or entirely different customizability options.

System Usability Scale

As this is the second iteration, there were hopes for a greatly improved SUS-score as it was the next iteration of the prototype. A higher SUS-score was reached, however, not to a particularly significant degree. This may be because the last SUS already scored very high and that the testing of domain familiar users could, in general, better understand the purpose of the designed feature. This could have given them a higher resolution of accuracy as to for example answering how often they would use the program. Compared to the paper prototype users that typically would answer either 1 if they thought they never use it, or 5 if they thought it might be useful.

Despite not reaching a great increase in the SUS-score on average compared to the previous iteration, these test results showed an overall lower variance for most of the SUS-questions. In addition to lower variance on the questions themselves, there was also a substantially lower standard deviation and variance on the average SUS-score as well. This can be

	Paper prototype	**Non-Func prototype**
Score 1	100	97.5
Score 2	92.5	95
Score 3	82.5	75
Score 4	67.5	87.5
Score 5	92.5	95
Score 6	97.5	90
Score 7	80	80
Standard Deviation	11.46	8.40
Variance	131.25	70.54

Table 7: Standard deviation and variance comparison of SUS-score of first and second iteration

seen in Table 7. These results were more *"stable"* than in the previous iteration. This prototype scored a total of 88.5, and with an average above 90 defined as excellent usability *[103] (Bangor, Kortum, and Miller **2008**)*, would imply that there is still some room for improvement. Usability is never perfect, and one can always strive to be better.

8 Functional Prototype

This section will present the third and final iteration, namely the functional prototype. A natural last phase of this project is to develop and create something more tangible than a non-functional prototype. Both for our own sake as developers, but also for a more palpable delivery to Schlumberger. Not to minimize the significance of the work up until this point, but a functional *"proof-of-concept"* prototype would aid Schlumberger engineers in seeing how one can implement some of the more significant additions introduced in this project. This includes drag and drop, drag to scale, the creating of thumbnails, and an example of possible data flow. This prototype also enabled fixing some of the issues detected after evaluating the previous prototype.

8.1 Prerequisites

When creating a modern web-based software, one picks certain frameworks and libraries to aid the process. This usually begins with the choice of the base framework for the front-end. The framework decision has to be based on assessing the importance of its learning resources, core features, popularity, usability, and of course the developers' familiarity. The three largest and most popular front-end frameworks today are Angular 2+ , ReactJS, and Vue.js. At the time of writing, ReactJS is the most popular of the three and is maintained by Facebook. It differs from the two in that it relies on additional libraries to function comparably. It does not aim to be a complete application framework. ReactJS is, as the name implies, written in JavaScript. Angular is the older and more feature-complete framework. It is maintained by Google, and it supports the usage of static typing as it is written in TypeScript. Vue.js is the least popular of the three, but it is rapidly growing in popularity. It is entirely open-source based, meaning no large company actively maintains it. It started as a lightweight project based on core ideas from Angular, and have ended up as a framework with the most in common with ReactJS. As with ReactJS, Vue.js is also written in JavaScript.

One can essentially build the same type of application with each of the aforementioned frameworks. The differences between them are usually not in any way essential in the making of the application. These frameworks are in functionality more or less interchangeable. Personal preferences are perhaps the first priority when choosing. In our case, coming as outsiders to an existing project where the framework was already chosen, it was only natural to use the same framework. It was, therefore, decided to use Angular 2+ as the framework for the last and final sprint of this project. By using the same framework as used in DrillPlan we would deliver a product with a higher value and we could expect more help from the developers at Schlumberger. It was also an opportunity to learn a new framework, since neither of us had any previous experience with Angular 2+ .

With the framework decided, we could continue with finding suitable libraries to ease the workload. As a large part of the functionality that was needed for the creation of a functioning prototype relies upon the action of dragging and dropping, we wanted to find

	Pros	**Cons**
Shopify Draggable	Has the potential to extend the built-in functionality. Good examples and demos online. MIT license	Not as easy to get to function properly as other frameworks. Messy documentation. Generic name → harder to debug.
Muuri	Easy to work with. Good examples. Good documentation. MIT license	Depending on how we want to implement the software, it may lack certain needed functionality. No "snap-to-grid" functionality. Is used more as a "sort elements from left to right" and not to place in a canvas
Dragula	Good name for searching. Quite easy to work with. MIT License	Only one direction of drag and drop at the time (either horizontal or vertical). Very basic, few ways of extending. No snap to grid. Not as nice to look at as other frameworks.
Gridstack	Promising examples and demos. MIT License	Based on Jquery, Initial problems of implementing it with Angular 2. Not a very active community behind it, meaning, limited functionality and support.
Sortablejs	Superior in popularity compared to the other drag'and'drop libraries compared [105]. Angular integration from the get-go. Good set of functionalities. MIT License	Saving the state of the stored elements can seem counterintuitive at first glance. No "snap-to-grid"

Table 8: Comparison of potential drag and drop libraries to be used

a library that fitted the closest to all of our specific requirements. This included having vertically sorted elements with variable size and the possibility to have special cases for modules like the header module. Like with the choosing of the frontend framework, it is essential, especially for us with less experience, that the library has good documentation and support, as well as being easy to use. The comparison of our selection of drag and drop libraries can be seen in Table 8. A separate GitHub branch was created for each of the drag and drop libraries, to see how difficult it was to set up a simple drag and drop interface and to explore the syntax and functionality of each. It was also crucial that any libraries used in this project were licensed under a permissive license, such as the MIT license, as to not limit its usage in a closed source environment like DrillPlan.

With these requirements in mind, the choice came down to either Muuri or Sortablejs.

In the end, Sortablejs was chosen above Muuri since it was more commonly used, which could mean easier debugging. Sortablejs did initially show issues with regards to saving the state of the items in a sortable list, but it was quickly sorted out. Like many other libraries, it did not support a *"snap-to-grid"* functionality. However, the inclusion of this functionality in the prototype was later dropped anyhow. Both since the feature did not seem as important as first believed, because the libraries give you the necessary feedback already. And also because of the additional work that the implementation of this feature would have implied, as few libraries supported it out of the box.

As DrillPlan is a multi-layered and complex software, its data and how it is stored is equally multi-layered and complex. As this prototype was never meant to be integrated directly into DrillPlan, there was not a need to spend time on backtracking and understanding the back-end of DrillPlan. But, since DrillPlan's data were queried through API's, and since API-calls can be seen with the help of e.g. Chrome Developer Tools, the potential of using these for the prototype were present. It meant having a larger data set available for the prototype, as each API returns large JSON-files with hundreds of lines of data. However, as the structure of these JSON-files proved to be quite overwhelming, and since we experienced issues with regards to authentication tokens, we decided to drop the use of API calls directly in our prototype to access the necessary data. Instead, data from an already created Well Montage was used[20], as it in and of itself contains several fields of data, and turned out to be adequate for a prototype of this kind.

To access the data that was extracted from the Well Montage, we needed some type of back-end to query this data from. Through previous positive experiences, Firebase[21] was chosen for this purpose. It is quick and easy to set up, and has several useful functionalities available out of the box, such as real-time database, authentication, and hosting. All these features are available for free if one's usage is below a certain quota, which the scale of this prototype would likely not reach. Firebase is also owned by Google and has an official library for combining Angular and Firebase, called AngularFire[22]. It makes for very few lines of code to get your website up and running with realtime querying of data that is secured through proper authentication.

8.2 Prototype Description

The functional prototype continued the usage of the facade introduced in the non-functional prototype. This was done by first creating a surrounding layout that looked and felt like DrillPlan. Meaning, a top-bar with the logged in user's name, and a sidebar to choose between the different sections of DrillPlan. As the current implementation of Well Montage is located in the *"Plan and Validation"* tab in DrillPlan, this is where the actual prototype would be located. The other buttons all redirected to the same empty page. Before getting access to the tab containing the Well Montages, the user has to log in with their Google-

[20]This was mocked data from a test project, meaning it was not from an actual well.
[21]`https://firebase.google.com/`
[22]`https://github.com/angular/angularfire2`

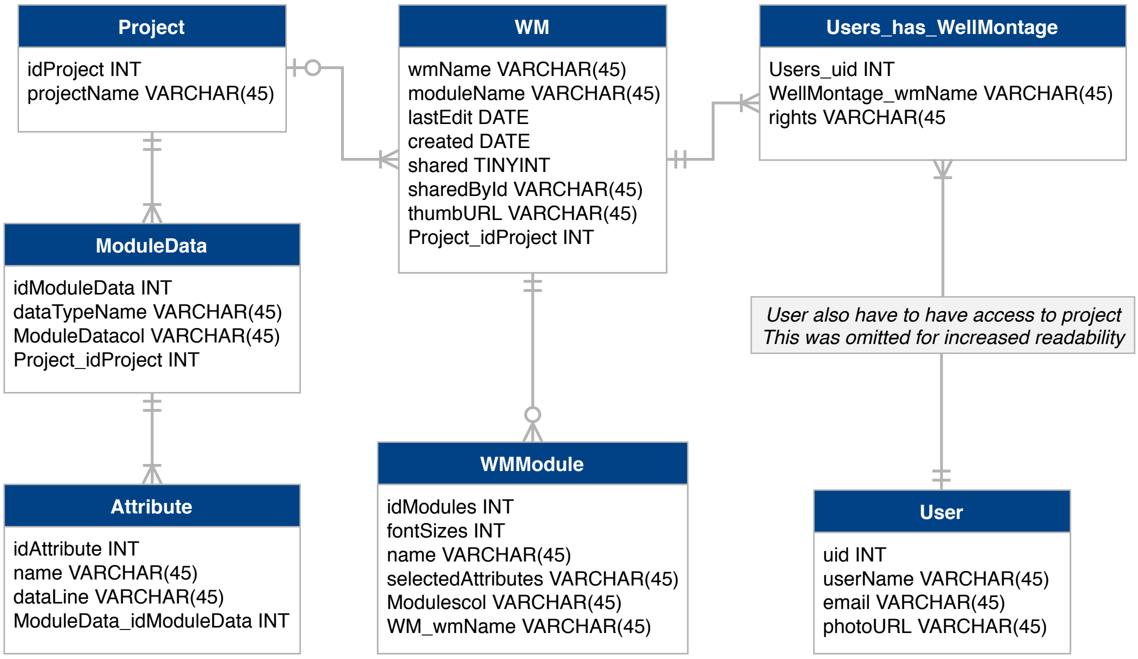


Figure 27: Database Entity Relationship Diagram of data on Firebase for the functional prototype

account. By specifying the authentication to only allow login via Google, and modifying the access rules of the data that is situated on Firebase to only allow the specific user ID of the writers of this paper, all data is secured from any potential penetration. Again showing the benefit of using Firebase - very minimal amount of work, to securely store and access data from a cloud storage. Figure 27 shows an ER-diagram of the planned database architecture for the functional prototype. The Firebase architecture differed by being a NoSQL database, meaning, the data structure is organized in a different way than traditional entity-relationship databases.

Main Menu

After logging in, and choosing the appropriate tab from the sidebar, the user is greeted with the main menu, which has not seen any major visual changes from the previous prototype. It can be seen in Figure 28. The orange icons that were previously used to visualize the share-state of the Well Montages, as seen in the previous prototype, has been removed. As the important information of who shared it was *"hidden affordance"*, meaning it was not entirely clear from the get-go as to where to obtain this information, the icons themselves were removed entirely. This information is instead showed directly beneath each Well Montage which is *"shared by others"*, with the caption *"Shared by"*. As seen in the figure, newly created Well Montages will get a thumbnail with their title. Well Montages that has been worked on will obtain a thumbnail, which previews what the Well Montage looks like.

The concept regarding templates and the template browser has been removed entirely from this prototype. Both due to time constraints, but also due to the additional complexity

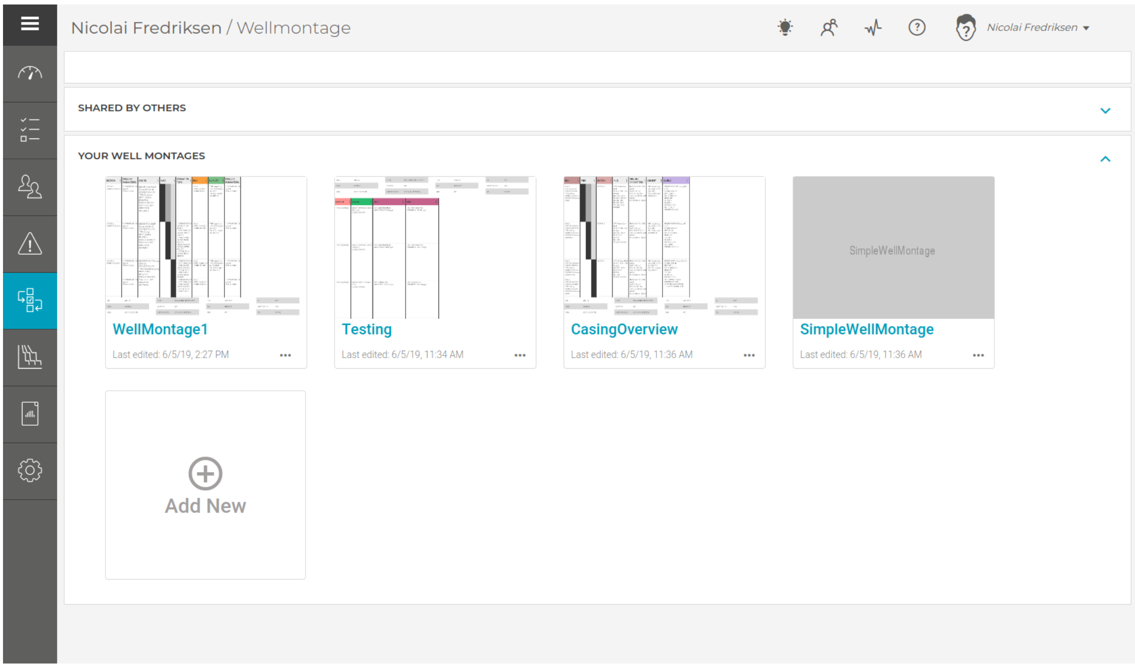


Figure 28: Main menu of the functional prototype

needed to implement such a feature in the prototype. The concept itself is rather substantial, and a further discussion regarding the strengths and weaknesses of the idea of templates and the sharing of templates can be found in the discussion on page 84.

Editor

The editor as well looks quite similar as to what it looked like in the previous prototype. There are some features on the top row that is disabled, again due to time constraints. Apart from that, most of the functionality that was tested in the previous prototype is present. Modules can be drag and dropped from the list on the right-hand side, and dropped onto the canvas. When the user drags the module from the list onto the canvas, a *"shadow"* of the minimal width of the module will be shown. The possibility of adding a module by simply clicking on the specified module was not added. The additional information that was shown for each module on hover, was also not included. As there are no actual restrictive rules for adding specific modules, the functionality was dropped. To reach the edit menu for a specific module, except for the header module, one clicks the three vertical dots for the corresponding module.

Some changes were done to the *"Save and Exit"* function from the last prototype. It did not convey a clear purpose in the non-functional prototype, as users pointed out. It was changed to be a button just for saving of the Well Montage. To exit the editor, the user can go back in the hierarchy either by using the sidebar, or the breadcrumb menu on top. Pulling inspiration, both from the feedback with regards to exiting with the use of the sidebar, and file explorers on operating systems which usually shows the hierarchy of files on the top row.

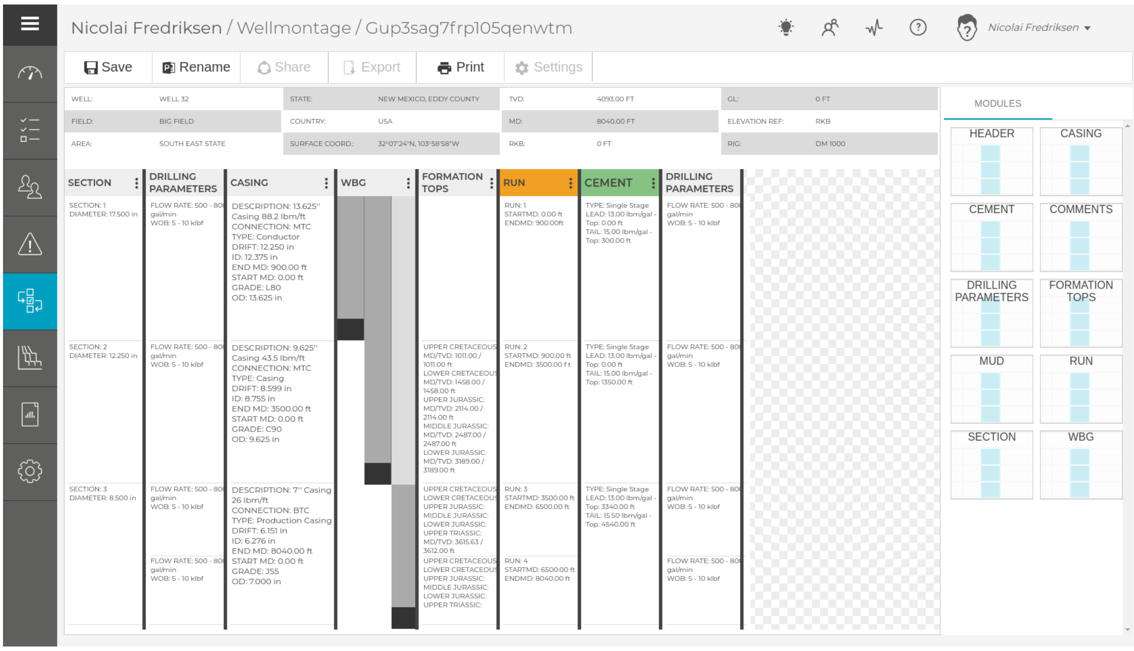


Figure 29: Editor of the functional prototype

Customizing Modules

A newly added feature is the potential of resizing modules' width. One does this by dragging on the vertical line on the right side of each module. To further enhance its affordance, the mouse changes to a resize icon on hover. This idea spawned quite early in the process of developing this prototype. After trying to place a module with some early sample data, we realized that the number of fields in a module might not fit in a fixed and limited module size. Thereby having a need for further customizability with regards to each module's size.

When entering a module's edit menu, the user can edit which of the available attributes that will be shown in the module. The live preview on the left-hand side, updated as the user added, removed, or sorted the attributes to their liking. There was also the possibility of editing some of the formattings for each module. To avoid the confusion that arose during testing of the non-functional prototype, there was added a *"ok"* and *"cancel"* button on the bottom right part of the window. With this change, we hoped that the user would be less reluctant to change attribute parameters, as these can be canceled.

By having this functionality, the editor makes each module quite customizable, apart from the header module which is a unique case. The header module in this prototype is quite unique in its nature, as it is static with the data that is shown, unlike the other modules. In addition, it can either be placed on top or on the bottom of the canvas, by dragging and dropping. This moves all the regular modules accordingly out of the way. A more modular header module is something that would have been nice to have, but its importance was deemphasized in favor of the other parts of the prototype.

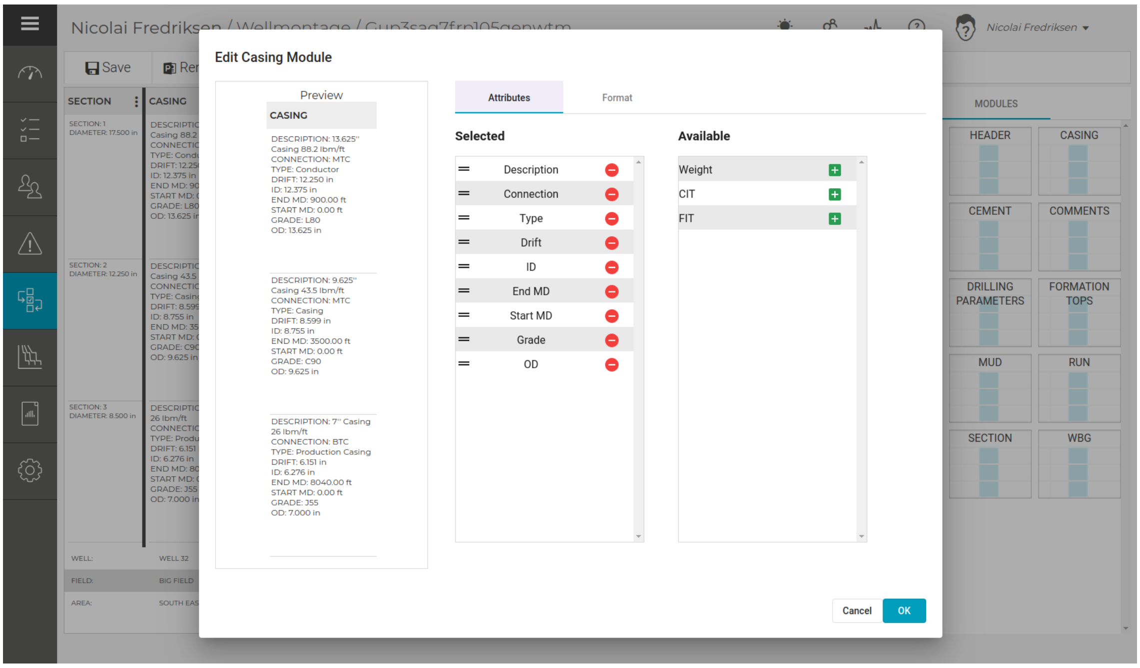


Figure 30: Edit module menu in the functional prototype

Architecture

Angular as a framework relies on what is called *components*. Which entails: separated parts of the interface is split into components, to separate logic for specific parts of the program. By splitting the logic to several components, changes to specific parts of the program are more easily done. The hierarchy of this prototype is as follows:

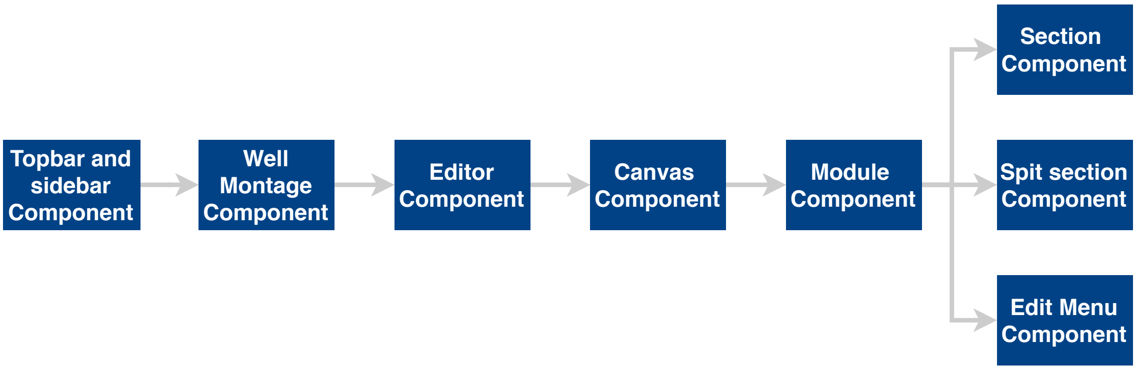


Figure 31: The component architecture of the functional prototype

- At the top level is the sidebar and the top-bar, as they are always present in the prototype.
- Below this is the Well Montage component for the main menu of the prototype. Here each user's specific Well Montages are loaded. And if the user picks a certain Well Montage, the corresponding data is passed down to the editor-component.
- The editor-component consists of the canvas component, and the module-list component. The module-list component reads all the module names from Firebase, and

when a module is dragged onto the canvas-component, a module-component is added to it.

- The ordering of the modules is based on the position of each module in an array in the canvas-component. Each module-component reads the corresponding data from Firebase, based on the module name. This includes module attributes, width, font sizes, and colors. The module is then rendered based on this data. Depending on if there is more than one run per section, the module-component either renders a regular section-component, or a split section-component.
- When editing a module, the data from that module is passed onto an edit-module, which is overlayed on top of the frame. The live preview is a regular module-component, with some additional parameters for rendering it inside the edit-component. Any potential changes done inside the edit-menu, are then passed back up into the corresponding module-component.

A simplified view of this hierarchy of components is illustrated in Figure 31.

8.3 Outcome and Aspirations

This prototype's source code and the master's thesis will be delivered to Schlumberger as is. Two presentations will be held for Schlumberger: one presenting and summarizing the work done in this thesis, and one presentation describing the architecture and implementation of the source code for this prototype.

As this was the final iteration, it was consequently done last. Meaning, as the deadline approached, things had to be wrapped up earlier than desired. Consequences of this were that not all features from the previous prototypes were implemented. A retrospective analysis of this and the other prototypes can be seen in Section 9.5. With regards to the missing functionality of this prototype, Section 9.6 describes the further work to be done. Also, the general stability and ironing of bugs were postponed, and the prototype may therefore not be of the desired standard and stability as one might want. However, the prototype shows the proof of concept of the configurable Well Montage clearly, especially by implementing the most complicated concepts. In sum, the iterations complete the goal of developing a configurable Well Montage.

9 Discussion

The discussion is divided into two parts. *Part 1* revolves around the first research question and looks with a macro level at the current state of the digitization of industries. It will also discuss what the future of the industry entails for both DrillPlan and Well Montage. *Part 2* discusses techniques and methods used in solving the case and will, therefore, discuss research question 2. It focuses on the explicit DrillPlan feature of Well Montage, and discusses its shortcomings and the process of creating an improved version.

Part 1

9.1 Industry Implications and Reluctance Towards Change

All industries are currently increasing the amount of data accumulated, and we are moving toward an ever more complex digital future. There is an increased challenge of successfully utilizing the digital resources gathered, as they grow larger than what humans can manage. These resources need to be available and searchable, and the workforces need new tools for both analyzing and for gaining increased knowledge. The increase of data in the oil industry from the exploitation of unconventional oil sources, lower oil prices, fewer employees with an increased workload, and a constantly evolving technology, are all factors that are related as to how the future of the industry will unfold.

Industries that earlier had separated software from multiple suppliers, are now in the need for these software to be communicating with each other. A tendency, at least in the industry of AEC and machine engineering, is to incorporate the separated tools, created for different professions inside the company, into *one* coherent tool. This enables increased work flow in the company by bettering the communication between employees and by speeding up the sharing of their professions' data. It also makes the work flow much cleaner, as each data point exists only in one place instead of being stored on a multitude of programs in different formats.

There are challenges related to the immediate future of industry software. Adoption of new systems often relies on the users' motivation to use the new systems. This can be described as a psychological reluctance towards change. If a new system means learning new ways of doing things, then the users' motivation tends to falter, and thus the adoption rates worsens *[37] (Robey and Sahay **1996**)*. An example of how a change is not successful is with the QWERTY keyboard layout. It is currently the standard keyboard-layout on all computers and mobile devices. The layout originates from a necessity to avoid jamming of the individual hammers of a typewriter. Letters which was often used in conjunction was spread out so that jamming was less likely to occur. It was successful in its mission of avoiding jamming, but it also introduced slower writing speeds and has shown to have a steeper learning curve for beginners. Now that typewriters are a thing of the past, and consequently the problem of jamming hammers too, one would expect the world to change the layout to a layout that is quicker and more efficient *[106] (David **1985**)*.

The example of the QWERTY-keyboard layout exemplifies the reluctance to change as a problem for technological advances. On the other hand, it also shows how powerful an industry standard can be. If one controlled the rights for the QWERTY-keyboard layout, one would be in a unique situation in the control of the keyboard market. Being the first to break out of a known system would be very difficult. Say your company was the first to change to another keyboard layout, there would surely be challenges related to employee satisfaction, for a while, and certain difficulties with cooperating with new actors visiting your company. A change of sorts would have to be clearly introduced and motivated towards the employees, and proper training for the employees would probably also proven necessary *[62] (Orlikowski **1995**)*.

Looking at the QWERTY-keyboard as a system, one could see the strength of the system being dependant on the number of users it has. One could talk about the network effects of the system, and that the strength is not being the best in its class, but is the most frequently used. When introducing a new system that is dependent on having a large user-base to function properly, one encounter the problem of bootstrapping *[49] (Hanseth and Lyytinen **2010**)*. An example can be seen in the AEC-industry, where a new BIM-system that markets its value by being implemented across the whole industry and which promises a new and better workspace, showed to be almost impossible to deploy. It has been shown extremely difficult to form these changes in an unorganized industry, where a common effort among all actors is needed to bring real industry change. Especially when no one really is in charge of an industry-wide grand plan *[32] (Miettinen and Paavola **2014**)*.

One could look towards big governmental contractors in the hopes of them mandating new industry standards that could inspire the rest of the industry. As with UK's BIM level 2 mandate from 2016, forcing BIM to all new governmental projects. A mandate affecting only parts of the sector is maybe not enough for a whole industry to change. But it certainly helps the actors of the industry to familiarize them with the concepts it introduces, and it gets them used to its practices. Whether this mandate is enough for spreading the use of BIM, in particular, to all parts of the world remains to see.

So far in the oil industry, no mandate for an industry-wide collaboration system exists. A mandate may, however, become necessary if the industry can not come together for the benefits of all. Otherwise, one must rely on other initiatives that help the spreading of the new industry-wide collaboration system. One of such initiatives could be that the developers of a system create a product that initially only focuses on the local benefits of a company. Meaning, that they do not focus on implementing the industry-wide system features from the get-go. But with time, if this new system has enough users, expand its capabilities to enable the increased collaboration one can see in the concepts of digital twin or BIM level 2.

The oil industry's silo mentality, meaning their inward look on the sharing of information and resources, should perhaps be discarded in favor of a more collaborative nature between companies. Examples from other industries show that all actors would favor from this. One could see an increase in the safety and correctness of drilling, also from unconventional sources. The sharing of data would ensure more comparable data for the engineers and geologist that are making decisions on whether to drill or not. In addition, the concept of digital twin, which takes the whole lifespan of a product into consideration, is something the oil industry especially could make use of given its complexity and risk-focused nature.

9.2 DrillPlan's Contingent Strategy

DrillPlan's current strategy is to deliver a web solution to cover all that is well planning. It does this in a way so that every employee of a company that has purchased DrillPlan, can use it as the only platform - where all their data and calculations are done. The data is stored in the cloud, and the platform provides a service for cloud-based calculations and analysis. It is the first platform that aims to have the full coverage of well planning's intricacies available on a web platform. By buying the DrillPlan solution today, a company could expect a better flow of information and to have a better overview of each drilling project internally.

DrillPlan aims at improving collaboration internally on a company level. Meaning, problems that can be resolved internally by changing a company's routines and practices, and not its entire value chain of collaborating subcontractors and affiliated companies. The users of DrillPlan can thereby expect a company-wide collaboration, working on the same data-set. You could go from a company built up of mostly separated teams working independently on their own system, to a company where the teams collaborate directly on the same platform. This approach of designing a platform does not rely on the network effects, as with digital twin, because it exclusively delivers a solution to the individual companies. It delivers a structural leap of cooperation by moving the boundaries for collaboration, but not in an extent as seen in the concept of digital twin.

A continued focus on the individual customer's needs could make Schlumberger's reputation as a software deliverer further increase. By doing user-centered product development, they could be able to please their customer's needs while also establishing new industry standards - benefiting new customers, their own reputation, and their value as a company. Having experience in creating industry standards and knowing how customers want to interact will generate a competitive advantage in the years to come. In order to succeed the goal of getting the largest user-base possible, they would benefit from having the most user-friendly system out there. Therefore, implementing new systems with a user-centered approach, with the use of UCD and the principles and guidelines discussed in Section 4.5, would be necessary to ensure a user-friendly system. If Schlumberger has ambitions to become a new *de facto standard* within well planning, the system itself has to be of high usability. However, this is not the only and deciding factor. Implementation of a new system entails much more than just creating a user-friendly system, as discussed with the bootstrap problem and the silo mentality with regards to the reluctance to change. However, the usability and UX of the system still has to be properly done to ensure proper adaption.

> *Despite all these technological advances, the need for a quality user experience remains paramount. If anything, the importance of ensuring a positive user experience keeps increasing. Given the pervasive information overload, combined with the expectation that everyone is computer savvy, the onus on designing for a quality user experience is even more critical these days.*
>
> - Excerpt from The UX Book
> *[107] (Hartson and Pyla **2012**)*

Our proposed strategy for Schlumberger moving forward will likely be to grow DrillPlan to encompass the whole drilling process, and not only the planning phase. After their customers have experienced the benefits of having an all-encompassing platform situated in the cloud, they will be more susceptible to the idea of a system covering the entire drilling operation. Thus, customers are more motivated to accept a new system. Making the concept of change easier for both company leaders, and their employees. By expanding their product portfolio, and constantly making sure new products communicate with each other and bring with it a similar experience, there could be the possibility for a vendor lock-in effect to develop[23]. Customers could, if done right, be dependent on the ecosystem of Schlumberger's products, as they cover all parts of the drilling process.

Given a successful expansion like depicted in the previous paragraph, Schlumberger would be in a particularly good position of developing digital twin in the oil industry, as the literature suggests could be the next step of digitization. Digital twin is a step towards the aforementioned industry-wide collaboration with far fewer borders for cooperation among the actors. By having DrillPlan as a web platform, enables changes to be easily made to encompass the new features necessary for a digital twin paradigm to develop.

By already having a high number of users, a new and expanded DrillPlan can avoid a potential bootstrap problem that would ensue with the inclusion of digital twin. Meaning, the number of users is already in place, so when Schlumberger *"flips the switch"* by adding more collaborative features, the user-base is already there. However, for it to be a potential future with the use of digital twin, everybody got to be involved. Smaller players in the industry can be motivated by the larger actors already using the technology. A focus on getting the larger actors on board should, therefore, be a priority. The smaller players will then benefit from being on the same collaborative industry-wide system as the larger players, and thereby create motivation for change.

As the inclusion of digital twin proposes an iterative expansion of DrillPlan into being something more encompassing, an agile mindset seem fitting. Since the existing users will already have a system that they benefit from, regardless of the next expansion, they do not have to wait until enough users have joined in before the system is beneficial. DrillPlan is thereby already beneficial for users by its current set of functionality. By aiming for a potential future with a further expansion of the platform, Schlumberger will have something

[23]Comparable to Apple's ecosystem of products, as they all communicate easily with each other, and a change of one product would disrupt this communication.

to fall back on if a more industry-wide collaborative digital twin future is too optimistic to actualize.

9.3 Managing Information Overload with a One-Page Summary

The recurring theme of the complexity of well planning is closely related to the problem of information overload. We are approaching the peak of oil production, conventional oil sources are drying up, and the number of unconventional sources for oil production is increasing. With these sources' nature being more complex than conventional sources, they are creating a need for more analysis through the processing of more data per well. A complete report for an entire well plan can result in years of planning and millions of pages of documentation *[63] (Benjamin **2018**)*. This combined with the fact that technology makes it possible to capture an increasing amount of data for analyzes, leads to the problem that is information overload in well planning *[5] (Kerschner and Capellán-Pérez **2017**)*, *[6] (Li **2018**)*.

One initiative created to tackle the information overload in well planning is the one-page summary. According to our interviews, it is a widespread technique used across the industry. The types of use cases where the one-page summary applies varies, but they all serve as a means of supporting the creation and sharing of knowledge. One typical use case is to have it as the center of discussions in meeting by decision makers. It is here a known reference point for all of the attendees of the meeting and can be a powerful tool in order to conceptualize new ideas. For this application, the one-page summary supports all categories of the knowledge spiral. It serves the *socialization* purpose in it being the explicit reference in meetings for discussion. It serves the *externalization* purpose by the generation of new documentation from the discussion in these meetings. The *combination* acknowledges that new insights are generated by synthesizing information from many different sources, which is what the one-page summary is a product of - processed data and calculations. And finally, *internalization* by transforming the explicit information located on the summary to a new tacit understanding by the attendees *[22] (Nonaka and Takeuchi **1998**)*.

Other use cases for the one-page summary includes it being the *"recipe"* of the drilling operation on site. It is often printed out on large-sized paper and hung up on the wall. It serves as a common look-up table for the different parameters of the drilling operation. Thus, summarizing the abundance of data. Finally, the purpose of having a look-up table extends also outside of the drilling location. One interviewee stated that the one-page summary is used as a template for filling out jurisdictional applications. The parameters of which to fill in the application is conveniently located in the one-page summary for easy access. Thus, being a convenient way of avoiding the information overload from extracting the required data from the drilling program.

The lack of documentation regarding this widely used feature goes to show the industry's current state of cooperation and secrecy. It demonstrates that not even conceptually trivial features have established standards. The industry would have benefited from having a convention as to how the one-page summary should look and behave, with standardized parameters to include and consistent visual cues for boosting the readability *[26] (Nygren and Henriksson **1992**)*. Cross-company cooperation would be easier with a common look and the sharing of information communicated more easily. The one-page summary should, therefore, be efficient and effective in its display of the data that is relevant to the individual user *[12] (Bole, Powell, Rousseau,* et al. ***2017****)*. The implementation of a one-page summary in DrillPlan, under the name Well Montage, can to an extent help define what a one-page summary is to look like. By DrillPlan gaining a large user base, one could expect that many of the features of DrillPlan could become more explicit examples for creating *de facto standards* in the industry.

Part 2

9.4 Conceptual Issues with the Case

Customizing Modules

Figure 10 on page 40, proposes a more extensive way of customizing modules than what is implemented in any of the iterations of the prototypes. Currently, customizations to the modules are done through choosing and sorting of attributes, and by selecting elementary formatting options. However, if one considers the DrillPlan platform and its enormous amounts of data, there is a potential for further expanding the visualization available through Well Montage. Figure 10 illustrates how the users of Well Montage can create their own modules in code, built upon any data available in DrillPlan.

The concept of being able to expand Well Montage's set of modules is in and of itself easy to understand. However, how one exactly chooses to enable the users to do so is not as trivial. Figure 10 proposed a code editor, but having only this way of customizing modules would make the barrier of entry for small visualizations very high. The current implementation of customizability may be too limited. Especially when considering the visual variation of one-page summaries currently in use, as seen in Figure 4 on page 21. The solution may be a question of how to conceptually boil down all the data that is available in DrillPlan into a user interface. It should deliver on all users' needs, and one may have to establish a strategy of having different levels of customizability, depending on the users' different levels of expertise. How deep the levels available for the end user goes, have to be evaluated on the premise of user-needs, user competence, and internal development cost of the feature. It may be easier, cheaper, and a more viable solution to let Schlumberger engineers develop the most advanced and customized modules in-house and make them readily available on demand. Cost analysis combined with more research on user-needs is necessary to successfully asses this.

Sharing Templates

The initiative of templates, as seen on page 38 can be described as a starting point for the users of the Configurable Well Montage. They will choose a template as a base, and for some, this is the only degree of customizability they will need. Meaning, some users will further customize a template, but some will not. The new solution should, therefore, come with a set of templates, made by Schlumberger. They should be made by Schlumberger, since they have the required domain knowledge to purposefully make these, and should be made based on what the users might want from a Well Montage.

Implementing the previous section's concept of more customizable modules, the amount of potential customizability significantly increases. Consequently, the amount of possible Well Montages increase, compared to what is currently possible with our proposed solution. Thus, it may become relevant for users to share these further enhanced Well Montages with other users of the system. Both internally within the company, but also the potential of sharing them to other users from other companies.

Sharing templates internally in a company should be encouraged, as to reduce work, but also to increase the collaborative nature of the work being done. It also has the possibility to create visual standards for a Well Montage throughout a company. If the feature is introduced, this should be a natural inclusion in the workflow of DrillPlan. But as discussed earlier, if one wants to expand its potential, the template browser should be something more akin of the Unity Asset Store[24]. Meaning, templates should be shared across companies, for the bettering and increased knowledge of all. However, with the silo mentality and the general reluctance to change, it is a question whether or not sharing across companies would be realistic in practice. There is also currently a design limitation to the concept of sharing elements beyond the borders of a project in DrillPlan.

9.5 Reflections on the Prototypes

Thoughts on Using Paper Prototyping

The paper prototype was as planned a low-fidelity prototype with an emphasis on the core functionality of the program in accordance with Section 13, more specifically; features regarding the editor. It succeeded in its goal of having the users understand what the editor entailed, and the tests showed that the conceptual model of modules, which are customizable, came through to the test subjects. Thus reaching a mental model for the users which we aimed for with this prototype *[91] (Norman* ***2013****)*. By having two versions of the prototype, we could also early on identify which aspects of both had the most usability problems, making it worth its while by reducing the need for making larger changes in later iterations *[77] (Virzi, Sokolov, and Karis* ***1996****)*.

The paper prototype was purposefully made with a limited scope, based on the map of user requirements that were made, as seen in Figure 13. Thus limiting the prototype to have a minimized scope in order to focus on the editor. This was done intentionally, and it made for a proper iterative design process with regards to the editor. However, as discussed in Section 9.4, the initiative regarding templates was postponed as an addition until the non-functional prototype. Meaning, it did not get the same iterative treatment

[24]`https://assetstore.unity.com/`

as the editor, which also could be part of the problem as to why its implementation in the functional prototype became overwhelming. In retrospect, templates should perhaps have been a part of the paper prototype, so its inadequacy in later iterations would have been avoided. This might not inherently be a cause of the paper prototype - the scope of the user map might instead have been at fault. Thus, blaming the lack of templates on the paper prototype might not be entirely accurate.

However, the lack of testing of the template browser is easily seen as a problem in retrospect, but it is not necessarily a feature that could have been foreseen as a core concept at the time. Lessons learned by this could be to try to further explore what makes up the core concepts at an earlier stage so that the most important features could be tested properly. Thus making what now is seen as difficult core concepts more vital in early stages, and not postponing the testing due to their perceived lesser importance at the time.

Despite this, by having paper prototyping, one can quickly make and test aspects of a program. Its low cost to produce, which we utilized by making two paper prototypes, aided in making something quick and dirty, which is necessary at the early parts of a project. Paper prototyping should then be used when it can be, as long as one plays to its strengths, and avoid its weaknesses.

Figma as a Prototyping Tool

The use of Figma as a prototyping tool could be seen as both a positive and a blessing in disguise. It was a good tool for creating an appealing and quick design, as its features and functionalities enhanced the capabilities of the designer. Meaning, as a design tool it was beyond expectations, making the placement of elements and management of the designs the best experience we ever had. However, as a prototyping tool, it lacked essential features to an extent that it dictated the interactions possible to create for the interactive prototype. An example being the lack of feedback on mouse hover from interactions of the test user. Most of these small feedback actions were possible to achieve through some perplex workarounds, but it meant a lot of extra work, which limited its use to only where absolutely necessary.

By being limited in how the prototype was made, despite a nice looking design that appeared complete, meant for stricter guidelines in the testing that would follow. What Figma does best is static images with links between them. Thus, making a prototype with more advanced interactions meant simplifying these advanced concepts and interactions so to comply with Figma's built-in limitations. Creating more work than strictly necessary if we had used a more fitting tool. The non-functional prototype became a prototype with a focus on visual design. What might have been done instead, given the appropriate tool, was to focus on the flow of the different parts of the program, and thereby simplifying the polish of the visuals till next iteration. This would have meant for a shorter time to develop said prototype, which in turn could have meant quicker user feedback of the flow. Doing it this way would enable a longer time spent on perfecting and developing the aforementioned advanced interactions in a functional prototype.

In the end, would there have been a better tool for the job? Depends on what one would define as a better tool. An alternative way of doing things would be to use Marvel instead. Marvel would have been a better choice for creating the non-functional prototype as it offers more functionality for creating advanced interactions. Then we could have later used Figma in parallel when developing the functional prototype to enhance its design. This would have meant more intricacies while working with Marvel, as by previous experience it quickly becomes overwhelming with larger projects.

Thereby, there might not be an actual *"perfect"* prototyping tool as they all have their own flaws. What one can take from this is; the perfect prototyping tool might not be a thing, however, a prototype could be made by using several tools. The tools will have their limitations, thus limiting the prototype in that manner. It is after all just a prototype, and limitations should be expected anyhow. The question to have in mind while selecting a tool for the job today should be whether to focus on the design or the interactivity aspect. By virtue of being satisfied with Figma as a design tool, its value as this should not be understated.

Functional Prototype

Overall, the functional prototype got to a satisfying quality and contains most of the wanted core functionality. By implementing the functional prototype in Angular, we got to attain new experience and knowledge with a commonly used front-end framework. After demonstrating and presenting both the prototype and the source code to Schlumberger, they too were very happy with the result. Particularly with regards to how through iterative work and rapid prototyping, the configurable Well Montage went from just an idea to a functional prototype. Thus, the functional prototype proved its value to the people at Schlumberger as a commendable proof of concept when evaluating the potential of implementing a configurable Well Montage into DrillPlan.

The time spent on the functional prototype was perhaps not sufficient. As the hours spent on this project was tracked, calculations showed that about a third of the raw hour amount was spent programming. The consequence of this was a prototype with a lesser amount of variety and functionalities than initially intended. Thereby cutting short the proposed list of functionalities first proposed during the requirements engineering phase. In addition, if we were familiar with Angular, more time could have been spent on making the prototype, instead of learning the front-end framework used.

However, looking back, the lack of functionality might not be as bad as first suggested. The functional prototype succeeding in being a successful demonstration of the advanced and new interactions, previously not tested properly in the other prototypes. In addition, it serves the purpose of showing that these types of interactions are possible with Angular, and thereby proves the validity of the propositions made by the prototype.

Testing of the Prototypes

With regards to the testing of the prototypes, the user guidelines with the tasks given to the test users could, by some, be seen as too strict and straight-forward with too few aspects of exploration. Knapp writes about how more open-ended questions and tasks will often result in more honest answers and results *[76] (Knapp, Zeratsky, and Kowitz* ***2016****)*. However, it was decided early on that we were to conduct the testing in a more strict and controlled manner.

We chose to test the first prototype, the paper prototype, on test subjects with no domain knowledge. It was, therefore, not our goal to test the prototype's domain-specific content. It was not a test to evaluate whether or not the prototype solved the problems oil engineers currently have. The test was purely a test of the interaction concepts. An arena of which we could assess window placements and if the interaction decisions of drag and drop, in this case, was better or worse than the older ways of interacting. The goal of the test was, therefore, to be left with empirical documentation for comparing the two created paper prototypes. While a freer conducted test plan would maybe uncover more unconventional errors and been the basis for forming more new ideas, it would certainly be less effective in being subject for a direct comparison afterward.

Moving to the non-functional prototype, the user tasks became focused on testing the flow of the program. We were again quite strict in the plan we wanted the users to follow. This time to unveil errors in placement of the different elements and buttons. This kind of testing will not cover the user's whole thought process from a complex idea to execution. Meaning, for them to execute a plan for a complex task of multiple subtasks. This test would, however, have a focus on executing the individual subtasks making up the complex tasks. The focus thereby became testing the user's ability to finding a specific next step in a longer thread of tasks. I.e. we created the complex tasks and asked the user to do the individual subtask. Ideally, you would want the test user to do both complex task and minor subtasks, but in order to have a better empirical foundation for comparing the different users, we decided to focus on the subtasks. Typically if one wants to test the whole flow, you have to let the users have a good understanding of the program before the testing can begin. It is only then that you could ask the user to do complicated and prolonged tasks. This is why we were more specific with the tasks at the beginning of the test, and looser after the users had gotten more familiar with the application. This way we tested more of the whole flow in the end and we also got to test how easy the program was to learn from the beginning.

Naturally, with the final prototype, the functional prototype, it would have been logical to test the whole user-flow with users without any previous knowledge of the system. There was unfortunately not enough time for us to do this. A test of such could be done in a similar manner as we conducted the two previous prototype tests, but a better approach would have been to launch the program to a selection of real customers. By analyzing their use patterns and by incorporation feedback systems, one could get sufficient qualitative results in order to do minor modifications to the system. One can roll out an update to an increasing number of users after the program gets refined by their testing, ensuring that the program performs as expected.

With regards to the usability metric of SUS, it did not pose as a large aspect of the process, as we might have hoped for by its inclusion. SUS was not something either of us was familiar with, and thereby, its implications and impact in a project were unknown. Firstly, there were issues with regards to SUS and non-native English speakers. This could have impacted the score, and translated questions should perhaps have been used instead *[104] (Finstad **2006**)*. Secondly, the constructive feedback aspect of the SUS-questionnaire may have colored the results to be more positive since the test subjects had a bias with regards to being acquaintances to the creators of the program. If a similar round of testing was done on people unknown from the creators, and with an anonymous SUS-questionnaire, the results might have become more honest, and thereby more helpful as to measuring the actual usability of the system. Despite this, the prototypes showed high usability throughout the testing that was done. Thereby, the SUS-questionnaire functioned as a tangible metric for further clarifying the usability and a direct way of comparing the prototypes' usability.

9.6 Finishing Thoughts and Further Work

The question can be asked whether or not an appropriate result was reached for this project, according to goals and expectations that were set. The functional prototype contains all the main functionality that makes it a compliant result as to what was expected. In addition, what we saw as the most important and challenging new aspects to be introduced with this solution, was implemented in the functional prototype. Thus, moving the focus away from smaller features and finishing touches.

A more exploratory prototype, with a wider set of new initiatives which could have made for a more interesting and not *"by the book"* end-result, was not something we strived for or even considered. This might come from what Knapp describes as *"the trouble with good ideas"*. He writes "*What's the most important place to focus your effort, and how do you start? (...) And how do you know when you've got the right solution? (...) And, once it's done, will anybody care?*" *[76] (Knapp, Zeratsky, and Kowitz **2016**)*. The focus might have been in the wrong place, as the solution might seem conventional, and not new and innovative. This might also come from a status-quo bias, from both us and Schlumberger, meaning favoring the current situation, which meant going for a solution that is more familiar in its execution.

The project began with the paper prototype trying to accept a design's viability, instead of trying something truly unique and interesting. Thereby, already from the start locking the product into a path of the familiar. Since the project tried to reach a certain goal by having a finished and tested solution at the end, the project as a whole may have been limited in the exploratory nature that might otherwise have introduced truly unique ideas. Instead, we went for a more strict timeline, with an emphasis on iterative prototyping. Thus, limiting the possibility of exploring the unknown.

> *Some people say, "Give the customers what they want." But that's not my approach. Our job is to figure out what they're going to want before they do. I think Henry Ford once said, "If I'd asked customers what they wanted, they would have told me, 'A faster horse!'" People don't know what they want until you show it to them. That's why I never rely on market research. Our task is to read things that are not yet on the page.*
>
> - Steve Jobs
>
> *[108] (Isaacson **2011**)*

Building on Steve Jobs' ideas, maybe the configurable Well Montage directly solves a problem that well planners are currently facing. The solution that we reached was a satisfying end-result, as it is the logical next version of Well Montage. Both as we were pleased with it, but also as it seemed to be a commendable solution based on Schlumberger's feedback. However, it might be something entirely different the users of DrillPlan actually need. The engineers that are planning a well wish to get rid of the manual and repetitive task of creating a Well Montage summarizing their joined effort, which includes the task of making sure it is always up to date with the latest data that is available. Perhaps the project premise itself, with a future where digital twin is more prevalent, has the wrong focus. Maybe the future enables an application that has a more dynamic approach through other mediums. Maybe the solution is to have something interactively available on your screen at all times? Not by doubting the actual legitimacy of the problem depicted by the users, but rather in an effort to progress toward the pipe dream solution that might be something more grand than *"just dragging and dropping modules onto a canvas"*.

Further Work

As the last iteration was *"cut short"* with previously tested functionality not implemented, there some further work to do on that front. Most of the additions are already defined as requirements in Appendix D, however, some of the ideas that either was not prioritized for functional development, or tested all together during any of the prototypes, will be mentioned here.

Firstly, the idea of exporting the Well Montage to another file format. This is something that will ensure backward compatibility, and make the transition over to the new Well Montage easier. In addition, it will comply with the need of accessing the Well Montage when having no internet connection. Thus, making it easier to access by having a local file. Its future necessity could be questioned, but it is a needed addition as of the industry state today, which the current implementation lacks.

The concept of templates and sharing of templates is yet to be added. The notion of sharing is an already established concept in DrillPlan, as sharing of workspaces and files is a recurring event in DrillPlan. However, as to how DrillPlan is currently designed, it only allows for sharing internally to people who are in the same project as you. For the idea of sharing Well Montage templates to reach its potential, there's a need to expand to the possibility of at least enabling sharing to colleagues outside of the project. It should be said that the goal of sharing across companies could be unrealistic with DrillPlan's current architecture. It is, however, something that Schlumberger should strive to introduce, as it would introduce industry-wide cooperation. The fully implemented sharing feature will yield more templates available, that potentially increases the quality of Well Montage and could possibly lead to an industry Well Montage standard.

For Schlumberger, there will also be further work of incorporating the functional prototype into today's DrillPlan. Aside from stabilizing the code and bug fixes, one essential change includes making use of DrillPlan's API's instead of Firebase for storing and accessing data. This could lead to an extensive rewrite of the code base that is delivered. In addition, the design of the functional prototype may not be completely in line with DrillPlan's already established design. Thus it is also the need to make it compliant with DrillPlan's design guidelines.

After the envisioned launch of our version of Well Montage, Schlumberger should see if there is a need for the *"custom modules"* feature, as introduced in Section 5. This is a quite expansive feature, that in short will enable users to create their own modules. For the users to be able to successfully use the feature, they will have to have knowledge of software development. The implementation of the feature will require additional testing and marked analysis from Schlumberger. They will have to research whether the amount of work that is needed to implement the feature is worth it, considering the small number of users that will be likely to make use of it.

With the design inspiration in mind, as seen in Appendix E, there are several other features that a more interactive solution could make use of.

- Snappable grid in the canvas, when adding and moving modules. This could enhance the user experience.
- Status showing the save-state of the Well Montage. Meaning, to know if your latest changes are saved or not. With the possibility of adding a commit message when saving, to know what a save entailed.
- History showing changes of the Well Montage. Both when, and who did each change.
- A clearer way to display errors if e.g. a module's font size is too large to display all the selected data fields inside the module.

In addition to these points, with the proposed initiatives of an interactive Well Montage, as seen in Section 5.2, it could be viable to further enhance the interactivity of the editor. Thus future-proofing for when a web view of the Well Montage is more susceptible to be used, in favor of a static Excel-like experience or a printed document.

As discussed earlier, our solution may be too *"conventional"*. Perhaps, a more unconventional and out of the box solution may be preferable with a future with the digital twin paradigm. There might even be a potential of even more efficient and effective web platforms, with new and upcoming tools and technologies such as WebAssembly[25], block chain, progressive web apps and machine learning.

[25] `https://webassembly.org/`

10 Conclusion

There is no questioning that the intricacies and complexity in well planning could make use of new emergent ways of thinking as to better the industry as a whole. DrillPlan is in a unique position with its all-in-one platform. With this in mind, Schlumberger could through an iterative deployment of their software be on top of new trends and paradigms in order to improve the world of well planning, both to their and their users' benefit. It is a tricky position, as there is no clear way forward, but as discussed, the potential for expansion towards an industry-wide collaborative set of features is certainly there. An abundance of data is not unique for the oil industry. However, as there are fewer mandates and external factors to be had, unlike other industries, Schlumberger is in a position to be proactive with regards to benefiting the industry as a whole. More collaborative and open efforts are seen elsewhere, so why not employ similar ways of thinking in the oil industry as well?

The one-page summary can be established as being in widespread use, and its uses are through our research exemplified and put in context. The case of the one-page summary and its use is so widespread, that one can wonder why it is still no standardized way of its use. It is a helpful technique used in well planning, both with regards to its ability to create knowledge and for the avoidance of information overload. In the ever-growing expansion of DrillPlan, with Well Montage, Schlumberger could propose new *de facto standards* by being the forerunner of a standardization process for the one-page summary. Thereby, benefitting everyone by unifying an already complicated and silo-mentality based environment, through more collaborative and zealous systems.

Our prototypes envision the next logical iteration for Well Montage. The development of our case constitutes a demonstration of how one can through iterative prototypes, with a user-centered focus, create a satisfactory and sufficient Well Montage with more configurability. The solution is a sum of tested and conceptualized initiatives presented throughout the multiple prototypes. It succeeds in creating a feature more in line with the nature of DrillPlan's dynamic web-platform. An implementation and release of our solution in DrillPlan will cover the users' current need for more customizability and will assist in handling the information overload of well planning. It will also facilitate in the way moving forwards towards a more collaborative industry if the initiatives of custom modules and sharing of templates get implemented.

11 References

[1] R. Frodeman, “Geological reasoning: Geology as an interpretive and historical science,” *Geological Society of America Bulletin*, vol. 107, no. 8, pp. 960–968, 1995.

[2] P. G. Almklov and V. Hepsø, “Between and beyond data: How analogue field experience informs the interpretation of remote data sources in petroleum reservoir geology,” *Social Studies of Science*, vol. 41, no. 4, pp. 539–561, 2011.

[3] R. Mitchell, L. Lake, and S. of Petroleum Engineers, *Petroleum Engineering Handbook*. Society of Petroleum Engineers, 2007, ISBN: 9781555631291.

[4] M. K. Hubbert *et al.*, “Nuclear energy and the fossil fuel,” in *Drilling and production practice*, American Petroleum Institute, 1956.

[5] C. Kerschner and I. Capellán-Pérez, “Peak-oil and ecological economics,” *Routdlege Handbook of Ecological Economics: Nature and Society. Abingdon*, pp. 425–435, 2017.

[6] D. M. Li. (2018). World oil 2018-2050: World energy annual report (part 2), [Online]. Available: `http://peakoilbarrel.com/world-oil-2018-2050-world-energy-annual-report-part-2/` (visited on 04/15/2019).

[7] J. Bradley, J. Loucks, J. Macaulay, A. Noronha, and M. Wade, “Digital vortex: How digital disruption is redefining industries,” *Global Center for Digital Business Transformation: An IMD and Cisco initiative*, 2015.

[8] U. B. of Labor Statistics. (2019). Employment, hours, and earnings from the current employment statistics survey (national), [Online]. Available: `https://data.bls.gov/pdq/SurveyOutputServlet` (visited on 05/20/2019).

[9] Norskpetroleum. (2018). Historical production, [Online]. Available: `https://www.norskpetroleum.no/fakta/historisk-produksjon/` (visited on 05/28/2019).

[10] R. Patterson. (2019). Was 2018 the peak for crude oil production? [Online]. Available: `https://oilprice.com/Energy/Energy-General/Was-2018-The-Peak-For-Crude-Oil-Production.html` (visited on 05/28/2019).

[11] B. Kolnowski, E. Ezersky, *et al.*, “Batch analysis tool-distribution system use cases,” in *PSIG Annual Meeting*, Pipeline Simulation Interest Group, 2016.

[12] M. Bole, G. Powell, E. Rousseau, *et al.*, “Taking control of the digital twin,” in *SNAME Maritime Convention*, The Society of Naval Architects and Marine Engineers, 2017.

[13] R. G. Dusterhoft, M. A. Strobel, M. Szatny, *et al.*, “An automated software workflow to optimize gulf of mexico lower tertiary wilcox sand reservoirs,” in *SPE International Symposium and Exhibition on Formation Damage Control*, Society of Petroleum Engineers, 2012.

[14] A. Al-Jasmi, A. Choudhuri, D. Joy, *et al.*, “Digital oilfield implementation in naturally flowing oil fields with increasing water encroachment: One company”s experience in burgan field,” in *SPE Intelligent Energy Conference & Exhibition*, Society of Petroleum Engineers, 2014.

[15] K. Mohaddes and M. H. Pesaran, “Oil prices and the global economy: Is it different this time around?” *Energy Economics*, vol. 65, pp. 315–325, 2017.

[16] C. Handscomb, S. Sharabura, and J. Woxholth. (2016). The oil and gas organization of the future, [Online]. Available: `https://www.mckinsey.com/industries/oil-and-gas/our-insights/the-oil-and-gas-organization-of-the-future` (visited on 11/26/2018).

[17] T. O. team. (2005). The psychology of decision making, [Online]. Available: `https://www.open.edu/openlearn/body-mind/psychology/the-psychology-decision-making` (visited on 05/17/2019).

[18] J. Gantz and D. Reinsel, "The digital universe in 2020: Big data, bigger digital shadows, and biggest growth in the far east," *IDC iView: IDC Analyze the future*, vol. 2007, no. 2012, pp. 1–16, 2012.

[19] P. I. Pecholcs, R. Al-Saad, M. Al-Sannaa, J. Quigley, C. Bagaini, A. Zarkhidze, R. May, M. Guellili, S. Sinanaj, and M. Membrouk, "A broadband full azimuth land seismic case study from saudi arabia using a 100,000 channel recording system at 6 terabytes per day: Acquisition and processing lessons learned," in *SEG Technical Program Expanded Abstracts 2012*, Society of Exploration Geophysicists, 2012, pp. 1–5.

[20] T. Jacobs *et al.*, "Shell, bluware team up on seismic with video game tech," *Journal of Petroleum Technology*, vol. 70, no. 10, pp. 42–43, 2018.

[21] S. R. Clegg and G. Palmer. Sage, 1996.

[22] I. Nonaka and H. Takeuchi, "10. a theory of the firm's knowledge-creation dynamics," 1998.

[23] E. A. Smith, "The role of tacit and explicit knowledge in the workplace," *Journal of knowledge Management*, vol. 5, no. 4, pp. 311–321, 2001.

[24] T. H. Davenport, L. Prusak, *et al.*, *Working knowledge: How organizations manage what they know*. Harvard Business Press, 1998.

[25] E. Monteiro, N. Pollock, O. Hanseth, and R. Williams, "From artefacts to infrastructures," *Computer supported cooperative work (CSCW)*, vol. 22, no. 4-6, pp. 575–607, 2013.

[26] E. Nygren and P. Henriksson, "Reading the medical record. i. analysis of physician's ways of reading the medical record," *Computer methods and programs in biomedicine*, vol. 39, no. 1-2, pp. 1–12, 1992.

[27] M. Cotsaftis, "What makes a system complex? an approach to self-organization and emergence," *arXiv preprint arXiv:0706.0440*, 2007.

[28] F. Tao, Y. Wang, Y. Zuo, H. Yang, and M. Zhang, "Internet of things in product life-cycle energy management," *Journal of Industrial Information Integration*, vol. 1, pp. 26–39, 2016.

[29] Aras, "A systems-centric approach for modern product development 5 concepts for engineering leaders," 2018.

[30] G. Sjøsåsen. (2017). Lundin norway present the benefits of aveva engage, [Online]. Available: `https://www.youtube.com/watch?v=zuLl3Ue7DLU` (visited on 05/21/2019).

[31] R. Miettinen, H. Kerosuo, J. Korpela, T. Mäki, and S. Paavola, "An activity theoretical approach to bim-research," in *A Paper at the ECPPM 2012 Conference*, 2012.

[32] R. Miettinen and S. Paavola, "Beyond the bim utopia: Approaches to the development and implementation of building information modeling," *Automation in construction*, vol. 43, pp. 84–91, 2014.

[33] D. Cao, H. Li, G. Wang, and T. Huang, "Identifying and contextualising the motivations for bim implementation in construction projects: An empirical study in china," *International journal of project management*, vol. 35, no. 4, pp. 658–669, 2017.

[34] A. Ghaffarianhoseini, J. Tookey, A. Ghaffarianhoseini, N. Naismith, S. Azhar, O. Efimova, and K. Raahemifar, "Building information modelling (bim) uptake: Clear benefits, understanding its implementation, risks and challenges," *Renewable and Sustainable Energy Reviews*, vol. 75, pp. 1046–1053, 2017.

[35] The Boston Consulting Group (BCG), "An action plan to accelerate building information modeling(bim) adoption," 2018.

[36] W. H. Bovey and A. Hede, "Resistance to organisational change: The role of defence mechanisms," *Journal of managerial psychology*, vol. 16, no. 7, pp. 534–548, 2001.

[37] D. Robey and S. Sahay, "Transforming work through information technology: A comparative case study of geographic information systems in county government," *Information systems research*, vol. 7, no. 1, pp. 93–110, 1996.

[38] S. J. Egan, *Rethinking construction: The report of the construction task force to the deputy prime minister, john prescott, on the scope for improving the quality and efficiency of UK construction; foreword by sir john egan.* Department of the Environment, Transport and the Regions, 1998.

[39] E. Chinyio and P. Olomolaiye, *Construction stakeholder management.* John Wiley & Sons, 2009.

[40] S. Parker, *McGraw-Hill Dictionary of Scientific and Technical Terms.* McGraw-Hill Education, 2003.

[41] Department for Business Innovation & Skills (BIS). (2019). About bim level 2, [Online]. Available: `https://bim-level2.org/en/about/` (visited on 05/20/2019).

[42] ——, (2019). Bim level 2 frequently asked questions, [Online]. Available: `https://bim-level2.org/en/faqs/` (visited on 05/20/2019).

[43] National Building Specification (NBS). (2018). National bim report 2018, [Online]. Available: `http://www.bimacademy.global/wp-content/uploads/2018/05/NBS0850-BIM-Report-2018-LR-.pdf` (visited on 05/20/2019).

[44] A. El Saddik, "Digital twins: The convergence of multimedia technologies," *IEEE MultiMedia*, vol. 25, no. 2, pp. 87–92, 2018.

[45] J. Post, M. Groen, and G. Klaseboer, "Physical model based digital twins in manufacturing processes," in *Form. Technol. Forum*, vol. 2017, 2017, p. 6.

[46] A. Frankel. (2015). Siemens plm - the real value of the digital twin, [Online]. Available: `https://www.youtube.com/watch?v=gK5sHDfBMP4` (visited on 05/16/2019).

[47] G. Rapier. (2019). Elon musk says building a factory is '100 times' as hard as building a car, [Online]. Available: https://www.businessinsider.com/elon-musk-says-building-factory-100-times-harder-than-making-car-2019-3?r=US&IR=T (visited on 05/13/2019).

[48] J. Van Os, "The digital twin throughout the lifecycle," *Maritime Convention*, 2018.

[49] O. Hanseth and K. Lyytinen, "Design theory for dynamic complexity in information infrastructures: The case of building internet," *Journal of information technology*, vol. 25, no. 1, pp. 1–19, 2010.

[50] A. Tiwana, *Platform Ecosystems. Aligning Architecture, Governance, and Strategy.* 2013, ch. 4, pp. 61–69.

[51] W. Roby, X. Wu, T. Goldina, E. Joliet, L. Ly, W. Mi, C. Wang, L. Zhang, D. Ciardi, and G. Dubois-Felsmann, "Firefly: Embracing future web technologies," in *Software and Cyberinfrastructure for Astronomy IV*, International Society for Optics and Photonics, vol. 9913, 2016, 99130Y.

[52] F. Shahzad, "Modern and responsive mobile-enabled web applications," *Procedia Computer Science*, vol. 110, pp. 410–415, 2017.

[53] G. Schroeder, C. Steinmetz, C. E. Pereira, I. Muller, N. Garcia, D. Espindola, and R. Rodrigues, "Visualising the digital twin using web services and augmented reality," in *2016 IEEE 14th International Conference on Industrial Informatics (INDIN)*, IEEE, 2016, pp. 522–527.

[54] K. M. Alam and A. El Saddik, "C2ps: A digital twin architecture reference model for the cloud-based cyber-physical systems," *IEEE Access*, vol. 5, pp. 2050–2062, 2017.

[55] Q. Chen, G. Reichard, and Y. Beliveau, "Interface management-a facilitator of lean construction and agile project management," *International Group for Lean Construction*, vol. 1, no. 1, pp. 57–66, 2007.

[56] C. Ferreira and J. Cohen, "Agile systems development and stakeholder satisfaction: A south african empirical study," in *Proceedings of the 2008 annual research conference of the South African Institute of Computer Scientists and Information Technologists on IT research in developing countries: riding the wave of technology*, ACM, 2008, pp. 48–55.

[57] I. Sommerville, *Software Engineering*, ser. International Computer Science Series. Pearson, 2011, ISBN: 9780137053469. [Online]. Available: https://books.google.no/books?id=l0egcQAACAAJ.

[58] G. S. Matharu, A. Mishra, H. Singh, and P. Upadhyay, "Empirical study of agile software development methodologies: A comparative analysis," *ACM SIGSOFT Software Engineering Notes*, vol. 40, no. 1, pp. 1–6, 2015.

[59] A. Cockburn, *Agile software development: the cooperative game*. Pearson Education, 2006.

[60] Atlassian. (2006). How o.c. tanner goes agile with jira service desk, [Online]. Available: https://www.atlassian.com/customers/oc-tanner (visited on 05/13/2019).

[61] A. Gostick. (2004). O.c. tanner named one of cio magazine's 'agile 100'; recognition company honored for combining it and enterprise agility, [Online]. Available: https://www.businesswire.com/news/home/20040823005661/en/O.C.-Tanner-Named-CIO-Magazines-Agile-100 (visited on 05/13/2019).

[62] W. J. Orlikowski, "Learning from notes: Organizational issues in groupware implementation," in *Readings in Human–Computer Interaction*, Elsevier, 1995, pp. 197–204.

[63] C. Benjamin, *Offshore drilling a recipe for disaster | city | halifax, nova scotia | the coast*, https://www.thecoast.ca/halifax/offshore-drilling-a-recipe-for-disaster/Content?oid=14800586, (Accessed on 12/04/2018), May 2018.

[64] J. E. Valusek, "Unocal thailand slashes well planning cycle time," *Oil & gas journal*, vol. 97, no. 2, pp. 45–48, 1999.

[65] SLB, *Schlumberger announces first-quarter 2018 results*, https://www.businesswire.com/news/home/20180420005288/en/Schlumberger-Announces-First-Quarter-2018-Results, (Accessed on 12/04/2018), Mar. 2018.

[66] S. Landmark, *Talisman-montney-ds-well-planning-case-study.pdf*, https://www.landmark.solutions/Portals/0/LMSDocs/CaseStudies/2012-07_Talisman-Montney-DS-Well-Planning-Case-Study.pdf, (Accessed on 12/01/2018), Jul. 2012.

[67] Schlumberger, *Drillplan*, https://www.software.slb.com/delfi/delfi-experience/drillplan, (Accessed on 12/10/2018), 2018.

[68] H. Choudhry, A. Mohammad, K. T. Tan, and R. Ward. (2016). The next frontier for digital technologies in oil and gas, [Online]. Available: https://www.mckinsey.com/industries/oil-and-gas/our-insights/the-next-frontier-for-digital-technologies-in-oil-and-gas (visited on 11/26/2018).

[69] K. Sadhwani, M. Sanderson, J. MacFarlane, S. Ibrahim, J. N. Garcia, *et al.*, "Drilling performance automation introduction in middle east asia," in *Abu Dhabi International Petroleum Exhibition & Conference*, Society of Petroleum Engineers, 2016.

[70] S. Rowe, S. Keasberry, J. Hunt, A. Arias, *et al.*, "The integrated environment for real-time decisions: Delivering excellence in brunei drilling operations," in *SPE/IATMI Asia Pacific Oil & Gas Conference and Exhibition*, Society of Petroleum Engineers, 2017.

[71] A. Al-Suwaidi, A. Faure, M. Talal, A. R. Al Taweel, A. R. Bamajbour, C. Edwards, J. Macklin, *et al.*, "Well delivery limit: Two years' experience with a breakthrough drilling performance initiative," in *Abu Dhabi International Petroleum Exhibition and Conference*, Society of Petroleum Engineers, 2002.

[72] A. Menhali, A. Mohammed, A. Sulaiman, J. S. Al Shamsi, B. Shamlan, A. Mubarak, A. N. M. A. Aishah, C. W. Kuyken, *et al.*, "One page programme (op2): New systematic drilling programme work flow for time and effort optimisation_a successful trial in adco," in *SPE/IADC Middle East Drilling Technology Conference & Exhibition*, Society of Petroleum Engineers, 2013.

[73] R. Atkinson, "Project management: Cost, time and quality, two best guesses and a phenomenon, its time to accept other success criteria," *International journal of project management*, vol. 17, no. 6, pp. 337–342, 1999.
[74] A. Sundvoll *et al.*, "Praktisk brukertesting," *Statistisk sentralbyrås håndbøker*, vol. 87, 2006.
[75] R. Einarson and N. Fredriksen, "Well montage: Investigating the oil industry's one-page well plan; its use, function and potential improvements," 2018.
[76] J. Knapp, J. Zeratsky, and B. Kowitz, *Sprint: How to solve big problems and test new ideas in just five days.* Simon and Schuster, 2016.
[77] R. A. Virzi, J. L. Sokolov, and D. Karis, "Usability problem identification using both low-and high-fidelity prototypes," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM, 1996, pp. 236–243.
[78] J. Preece, Y. Rogers, and H. Sharp, *Interaction Design: Beyond Human-Computer Interaction*, 4th. Wiley, 2015, ISBN: 9781119020752.
[79] J. Sauer and A. Sonderegger, "The influence of prototype fidelity and aesthetics of design in usability tests: Effects on user behaviour, subjective evaluation and emotion," *Applied ergonomics*, vol. 40, no. 4, pp. 670–677, 2009.
[80] M. Walker, L. Takayama, and J. A. Landay, "High-fidelity or low-fidelity, paper or computer? choosing attributes when testing web prototypes," in *Proceedings of the human factors and ergonomics society annual meeting*, SAGE Publications Sage CA: Los Angeles, CA, vol. 46, 2002, pp. 661–665.
[81] A. Sarma and A. Van Der Hoek, "Towards awareness in the large," in *2006 IEEE International Conference on Global Software Engineering (ICGSE'06)*, IEEE, 2006, pp. 127–131.
[82] J. Rubin and D. Chisnell, *Handbook of usability testing: how to plan, design and conduct effective tests.* John Wiley & Sons, 2008.
[83] J. Vijayan and G. Raju, "A new approach to requirements elicitation using paper prototype," *International Journal of Advanced Science and Technology*, vol. 28, pp. 9–16, 2011.
[84] A. Kleon, *Steal like an artist: 10 things nobody told you about being creative.* Workman Publishing, 2012.
[85] E. Toftøy-Andersen and J. G. Wold, *Praktisk brukertesting.* Cappelen Damm Akademisk, 2011.
[86] J. Nielsen and T. K. Landauer, "A mathematical model of the finding of usability problems," in *Proceedings of the INTERACT'93 and CHI'93 conference on Human factors in computing systems*, ACM, 1993, pp. 206–213.
[87] J. Nielsen, "Estimating the number of subjects needed for a thinking aloud test," *International journal of human-computer studies*, vol. 41, no. 3, pp. 385–397, 1994.
[88] R. A. Virzi, "Refining the test phase of usability evaluation: How many subjects is enough?" *Human factors*, vol. 34, no. 4, pp. 457–468, 1992.
[89] C. Snyder, *Paper prototyping: The fast and easy way to design and refine user interfaces.* Morgan Kaufmann, 2003.

[90] J. Nielsen, "Enhancing the explanatory power of usability heuristics," in *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, ACM, 1994, pp. 152–158.
[91] D. Norman, *The design of everyday things: Revised and expanded edition.* Constellation, 2013.
[92] J. Brooke *et al.*, "Sus-a quick and dirty usability scale," *Usability evaluation in industry*, vol. 189, no. 194, pp. 4–7, 1996.
[93] J. Brooke, "Sus: A retrospective," *Journal of usability studies*, vol. 8, no. 2, pp. 29–40, 2013.
[94] T. S. Tullis and J. N. Stetson, "A comparison of questionnaires for assessing website usability," in *Usability professional association conference*, Minneapolis, USA, vol. 1, 2004.
[95] D. Esposito, *Modern web development: understanding domains, technologies, and user experience.* Microsoft Press, 2016.
[96] J. Johnson and A. Henderson, "Conceptual models: Begin by designing what to design," *interactions*, vol. 9, no. 1, pp. 25–32, 2002.
[97] B. Shneiderman, "The eyes have it: A task by data type taxonomy for information visualizations," in *Visual Languages, 1996. Proceedings., IEEE Symposium on*, IEEE, 1996, pp. 336–343.
[98] J. Nielsen, "Iterative user-interface design," *Computer*, no. 11, pp. 32–41, 1993.
[99] J. Gulliksen, B. Göransson, I. Boivie, S. Blomkvist, J. Persson, and Å. Cajander, "Key principles for user-centred systems design," *Behaviour and Information Technology*, vol. 22, no. 6, pp. 397–409, 2003.
[100] J.-Y. Mao, K. Vredenburg, P. W. Smith, and T. Carey, "The state of user-centered design practice," *Communications of the ACM*, vol. 48, no. 3, pp. 105–109, 2005.
[101] I. Ershaghi, Z. S. Omoregie, *et al.*, "Continuing-education needs for the digital oil fields of the future," in *SPE Annual Technical Conference and Exhibition*, Society of Petroleum Engineers, 2005.
[102] M. O. Leavitt, B. Shneiderman, *et al.*, "Based web design & usability guidelines," *Background and Methodology*, 2006.
[103] A. Bangor, P. T. Kortum, and J. T. Miller, "An empirical evaluation of the system usability scale," *Intl. Journal of Human–Computer Interaction*, vol. 24, no. 6, pp. 574–594, 2008.
[104] K. Finstad, "The system usability scale and non-native english speakers," *Journal of usability studies*, vol. 1, no. 4, pp. 185–188, 2006.
[105] Npmtrends. (2019). Dragula vs interactjs vs sortablejs vs @shopify/draggable vs gridstack vs muuri, [Online]. Available: `https://www.npmtrends.com/dragula-vs-interactjs-vs-sortablejs-vs-@shopify/draggable-vs-gridstack-vs-muuri` (visited on 05/20/2019).
[106] P. A. David, "Clio and the economics of qwerty," *The American economic review*, vol. 75, no. 2, pp. 332–337, 1985.
[107] R. Hartson and P. S. Pyla, *The UX Book: Process and guidelines for ensuring a quality user experience.* Elsevier, 2012.

[108] W. Isaacson, *Steve Jobs*. Simon & Schuster, 2011.

A Master's Thesis Proposal

Well montage report is a one page document summarizing engineering details and general information about planned well. Well montage report template differ from company to company, thereby locking the report to a predefined template is not a scalable solution.

The idea of the project is to create interactive application that would allow well engineers to create custom well montage reports. The application would allow to choose one of the DrillPlan data sources and add an interactive widget to the well montage canvas. Widgets could be configurable allowing to choose specific data format/styling for a selected data source.

There are relationships between the data sources that would impose layout constraints. For example if BHA from a second well section is chosen, a layout should automatically scale into two well sections. The final output should be exported into printable document.

Exact interaction patterns and user experience details could be worked out through a user research. Access to a user base will be provided.

Project scope will include user research, user experience and visual design, feature prototyping and commercializing (optional).

B Interview Guide for Case Study

Interview-Guide

(Semi-structured)

1. Introduction: Who are we, what we study, purpose of interview - master thesis in cooperation with Schlumberger. New functionality planned, and want to expand this functionality based on feedback gained from the interviews that are conducted.
    a. Inform that notes will be taken during the interview
    b. No personal information, except your background and work-position and signature(name) on the interview-agreement, will be saved. You will be anonymous, and anything you say will not be traceable back to you
    c. Any questions before we begin?
2. About the interviewee:
    a. What are your background in the oil industry(what do you work with),
    b. Your (user's) use of DrillPlan, experience with other digital aids in the oil industry,
    c. Experience with digital tools in general (computer knowledge)
3. About Well Montage in its current state:
    a. What are your general thoughts on well montage,
    b. Who are the different users that is using Well montage (user-types)
    c. How do the different users use it in their work. Reference, main point of interest
    d. What are the most important parts of the output (which columns, fields, or the sheet as a whole etc)
    e. Anything you feel like are missing from the Well Montage
        - TVD vs MD, specific fields, layout, clearer headers for each column, "live updates", history of changes (who changed what and when)
    f. What about something that's not an excel sheet, but a different visualization of the well (e.g. more graphical with less text, 3D view of a well, Show correlation between data clearer, VR/AR etc)
4. About configurability:
    a. If you had the ability to modify how the output would look what would you like to add/remove.
        - Removal of entire columns, fields, adding of other columns with information that's not there already?
    b. Should this be "saveable" as templates so different configurations can be shared across the project/company?
        - How many different templates do you see yourself using, if any?
    c. Customizability vs several available "templates" of well montage
        - more advanced/difficult to customize, easier to choose between different premade templates
5. Summarization
    a. Summarize notes taken
    b. Anything you would like to add?
    c. Any misunderstandings?
    d. Any other general questions?

C Project Plan

	Week	Task
Phase 0	6	Setting up workspace
	7	**Writing** - Set up doc and literature
Sprint 1	8	Prototype 1, Paper prototyping with testing
	9	
	10	
Sprint 2	11	Prototype 2, wireframe
	12	Testing
	13	**Writing** - Document the prototype 2
Sprint 3	14	Programming
	15	Testing & assistance
	16	**Writing** - Document the prototypes
	17	Programming
	18	
	19	Final testing
	20	Finalization & cleanup
Finish	21	**Writing** - Master's thesis
	22	
	23	

D Sticky Notes Requirements

Figure 32: Categorized sticky notes with requirements from the "Be Able To" session

Theme	Description (Be able to)	Priority	Hours
Print	Click a button to instantly print the well montage (Bring up chrome print setup)	5	10
Print	Have a print friendly look (Regarding font sizing, it fits within the paper margins etc)	5	15
Templates	Save current version (Save the current setup of modules to a file)	5	15
Templates	Choose starting template/Use existing template (A template browser)	5	15
Formating	Format module with regards to fields, size (Click module, menu for editing a single module)	5	45
Formating	Be able to move every part of WM (Every component on the well montage should be movable	5	20
Formating	Choose between column-based and singular-based modules (Two menus for these two)	5	15
Modules	Remove modules from WM (Remove modules from the well montage)	5	7.5
Modules	Select and place blank modules (Basic selecting and placing of blank modules on a canvas)	5	7

Modules	Edit and format general information (name. Location. Info available from project in drillplan)	5	7
Misc	Create a new well montage within 3 minutes. (non-functional: 3 minutes from start to finish)	5	5
Templates	Re-open personal WM, which was previously worked on. (Find a saved WM, and open it)	4.5	15
Print	Scale to paper (Auto-fit and size information, with regards to the chosen paper size)	4	35
Formating	Change font and font color (For the whole well montage itself)	4	20
Changes	See change log (Show changle log, for changes made in the well montage, and who did them)	4	40
Changes	Auto incremental version number (A number which increments everytime a change is made)	4	5
Modules	Add custom text fields (A module for writing your own text in)	4	10
Modules	Place modules according to predfined grid (Every module should be placed within a e.g. 1cm grid)	3.5	15
Print	Print to file (Click button, and the well montage in the web is converted to e.g. PDF/excel. Today's functionality)	3	30
Templates	Share WM across company (internally and externally). Custom WM's is accessable to other people, if you choose so	3	35
Formating	Offer different visualizations e.g. regarding WBG. (Offer different types of visualizing WBG and other things that's relevant to visualize)	3	50
Modules	Easily drag-n-drop modules. (Modules should drag-n-drop from a menu, which supports tablets and a more user friendly experience)	3	20
Misc	Use tablets for creation (Future proof the well montage generator, to support the use of tablets)	3	10
Misc	Have multiple "active" WM (Active WM's category in template browser)	3	5
Misc	Use dropdown to create custom modules (A "scratch"-like way of creating custom templates)	3	20
Misc	Create custom modules (Custom modules from SQL-like queries)	3	90
Formating	Edit Modules color and look (Edit look of singular modules)	2	12

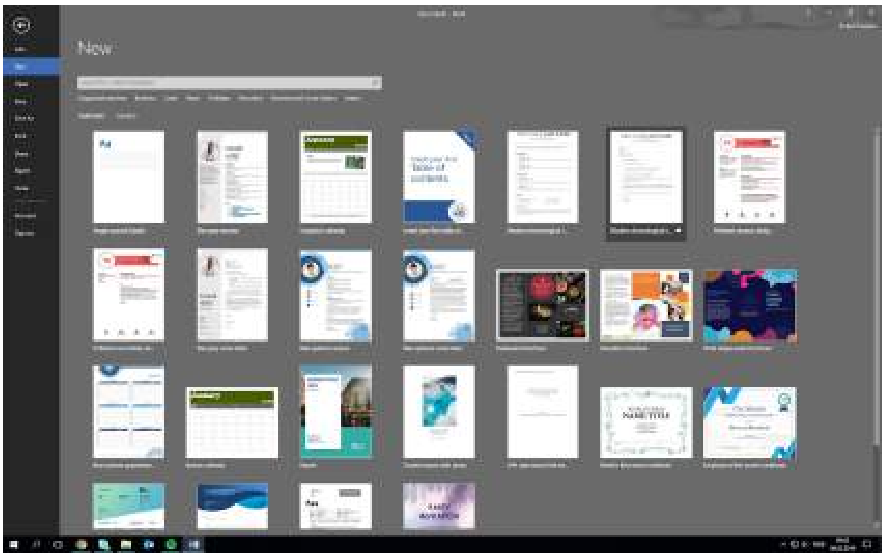
Formating	Get notified if font is too small (If font is to small with regards to the chosen paper size)	2	10
Changes	Comment and tag people anywhere (for changes) (Google drive like functionality, for commenting and tagging people within the WM)	1.5	60
Templates	Share my customization (Share color/look/feel of the WM to other people. Not the setup of modules)	1	25

Table 9: Digitized version of "Be Able To" session with; enhanced description, Priority ranking from 1-5 (With 5 being the highest), and estimated hours of development

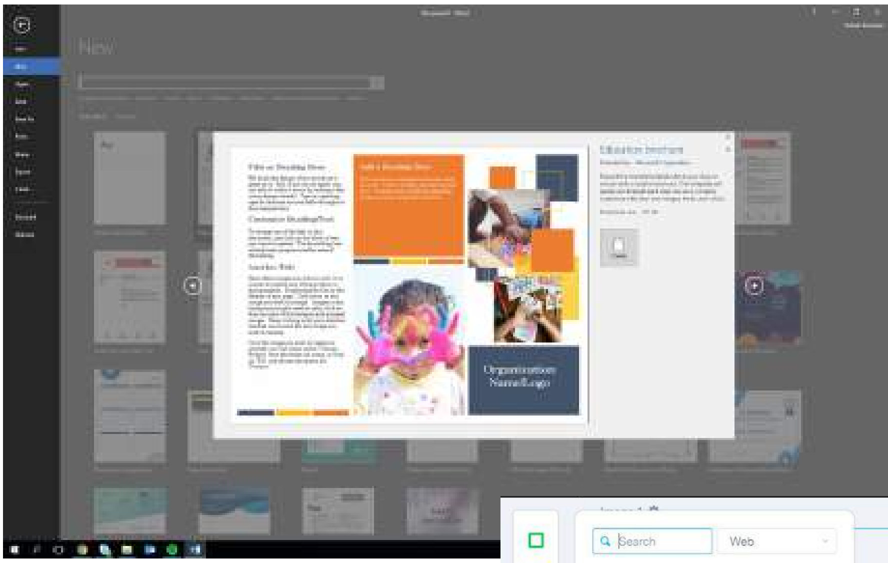
E Design Inspiration

(1) Select from set of ready-made templates:

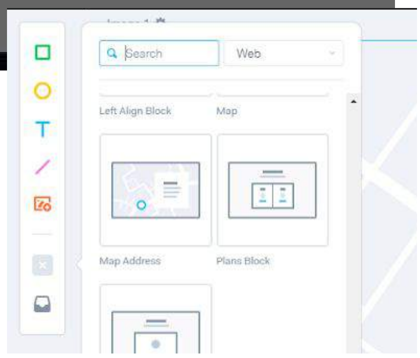
(2) Word distinguishes between Featured and Shared eg. “Shared” is on campus machines NTNU specific letters etc.:

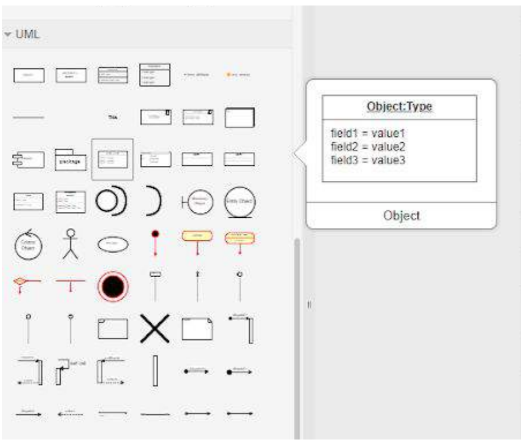


(3) One can also "preview" each template where the author and some info is hidden.



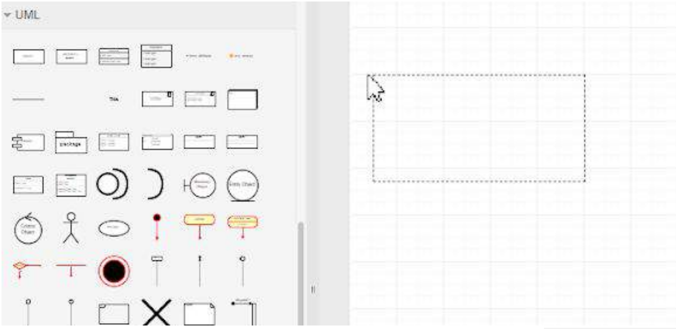
(4) Search with highlighted search box and single filter on the page:





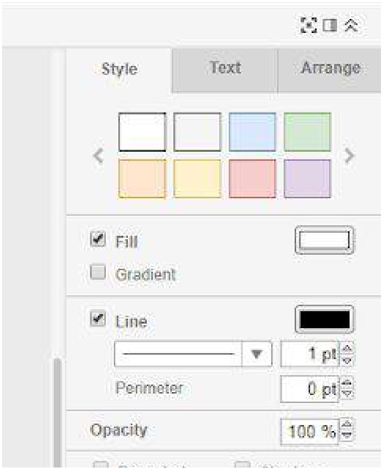
(5) Mouse over to see larger example widget:

(6) Build the canvas with snapable grid perhaps, if we are to support some form of scaling of modules:



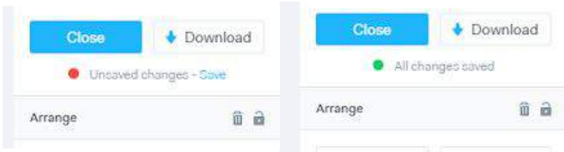
(7) Let the user both drag and drop, but also just click on widgets to add them to the canvas:

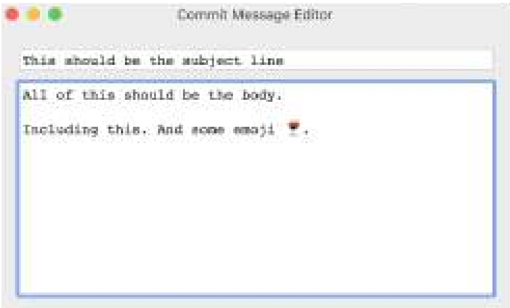
(8) Separate into different tabs per widget on Style, Data (++). Include warnings / issues / alerts here too? To see things that are wrong with the document (picture of this further down).



(9) Maybe one should have a plus symbol where a widget belongs, but this may yield too many interaction possibilities which can render the app difficult to use. Needs to be tested:

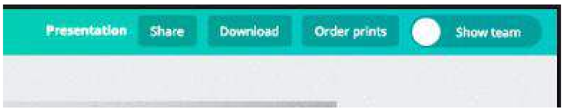
(10) Status for the document (unsaved, saved):

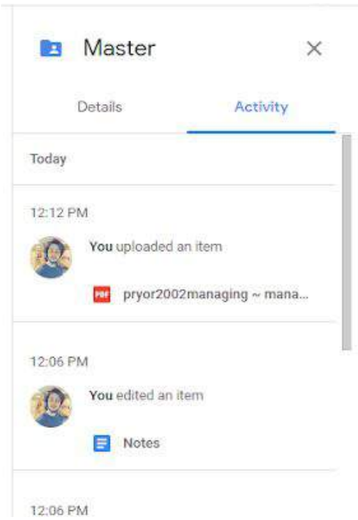




(11) Save should perhaps have to give a kind of commit message when saving?

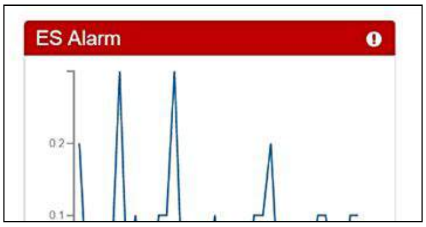
(12) Bar on top with print, publish, details (where to change size parameters etc?)



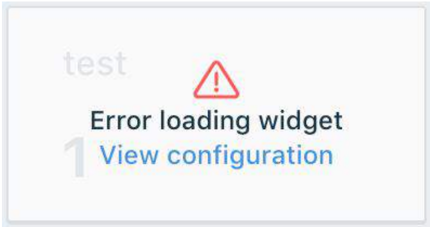


(13) Incorporate history for the document, possible drill plan already has a good way to do this. But drive has at least a little inspo under their "Activity":

(14) Widgets alerts if font is probably too small for the selected paper size. A little icon in the corner, preferably yellow triangle with exclamation mark. Mouse over to see text describing the problem:

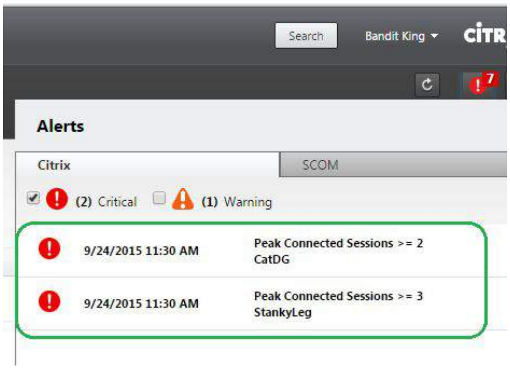


(15) Widget fails:



(16) Include one of these where changes are made, for example under Activity/History:

(17) Have a list with errors:



F Information Letter

Are you interested in taking part in the research project

Configurable Well montage dashboard

This is an inquiry about participation in a research project where the main purpose is to map and test user requirements regarding the Well montage dashboard in order to further enhance the product. In this letter, we inform you of the purpose of the project and what your participation will involve.

Purpose of the project
The purpose of this research is to map and test user requirements when using Schlumberger's Well Montage functionality. This to better understand and identify issues of the application in order to further enhance it. The research will be part of the basis in order to form an informed decision on which features to develop.

The research involving user participation will consist of observations in the the period from January to June of 2019. The project is done as a part of a master's thesis done in the spring semester of 2019 at NTNU.

The data collected from this research will only be used with regards to the master's thesis and for developing a working prototype.

Who is responsible for the research project?
The project manager is Eric Monteiro situated at NTNU Norwegian University of Science and Technology/ Faculty of Information Technology and Electrical Engineering/ Department of Computer Science, the institution is responsible for the project.

The master thesis is written by Nicolai Fredriksen and Robert Einarson, both studying Computer Science with a specialisation in Interaction Design and Game technology. In cooperation with the project's supervisor, we form the project members.

The project is also written with the assistance of Schlumberger Information Solutions Norway Technology Center, located in Asker.

Why are you being asked to participate?

a) **In association with Schlumberger:**
You have been selected either randomly from a selection of candidates within your company by us, on approval from your supervisor, or selected in cooperation with Schlumberger and your company. As this part of the study belongs to the qualitative research, you are only one of a selected few.

This information letter may be sent on our behalf by your supervisor. This in order to ensure optimal test candidates for this research.

b) **Outside of Schlumberger**
You have been selected based on your field of study, to accomodate for the needed technical background in the expected target groups knowledge of user interface navigation. You have been selected with regards to your location and affiliation to the project members. As this part of the study belongs to the qualitative research, you are only one of a selected few.

What does participation involve for you?
If you choose to take part in this project, you agree to be subject to usability testing of the Well Montage dashboard. The usability testing comes in multiple phases, and the exact method of testing will therefore differ as the project evolves. Common for all usability testing for the participants in this project will involve navigation through graphical user interfaces. You will be given some general information of what your participation involves and instruction on how you interact with the prototypes up for testing. You will also be asked to think out loud, this to make your intentions and reasoning behind decisions in the application clearer. A set of tasks will be given to you throughout the testing session, in addition to follow-up questions for additional clarification.

The test will take no more than 60 minutes to complete.

There may be audio or video recordings done of the session with your consent. These will not be published, and will only be used for referential purposes for the project members.

Participation is voluntary
Participation in the project is voluntary. If you chose to participate, you can withdraw your consent at any time without giving a reason. All information gathered from your session will then be deleted. There will be no consequences if you choose not to participate or later decide to withdraw your consent.

Your personal privacy – how we will store and use your personal data
We will only use your personal data for the purpose(s) specified in this information letter. We will process your personal data confidentiality and in accordance with data protection legislation (the n).

- The data collected in this research will only be accessible for the two students conducting the research and the responsible supervisor.
- Potential directly and indirectly traceable information gathered through the sessions will be kept in accordance to NTNU's rules updated for GDPR[1] for handling personal data in research projects. Written documents will be locked away with access only for the project members. Digital files will be safely stored with encryption and limited access in accordance to NTNU's rules.

[1] General Data Protection Regulation

- The published information will be anonymized and should not be directly recognizable or traceable to any specific users of the system.

What will happen to your personal data at the end of the research project?
The project is scheduled to end by the 30th of June 2019. By this date arrives, all personal data , both video/audio recordings and any potential physical or digital notes, will be deleted.

Your rights
So long as you can be identified in the collected data, you have the right to:
- access the personal data that is being processed about you
- request that your personal data is deleted
- request that incorrect personal data about you is corrected/rectified
- receive a copy of your personal data (data portability), and
- send a complaint to the faculty's NSD contact or The Norwegian Data Protection Authority regarding the processing of your personal data

What gives us the right to process your personal data?
We will process your personal data based on your consent.

Based on an agreement with NTNU Norwegian University of Science and Technology/ Faculty of Information Technology and Electrical Engineering/ Department of Computer Science, NSD – The Norwegian Centre for Research Data AS has assessed that the processing of personal data in this project is in accordance with data protection legislation.

Where can I find out more?
If you have questions about the project or want to exercise your rights, contact:
- NTNU Norwegian University of Science and Technology/ Faculty of Information Technology and Electrical Engineering/ Department of Computer Science via supervisor for the project Eric Monteiro by email: (eric.monteiro@ntnu.no) or by telephone: +47 73 59 67 51
- NTNU's own set of rules for collecting and storing of personal data for research projects (Only in Norwegian as of Feb. 2019): https://innsida.ntnu.no/wiki/-/wiki/Norsk/Behandle+personopplysninger+i+forskningsprosjekt
- The students running the project: Robert Einarson (roberei@stud.ntnu.no) or Nicolai Fredriksen (nicolafr@stud.ntnu.no)
- The faculty's NSD contact: Harald Lenschow by email: (harald.lenschow@ntnu.no) or by telephone: +47 73 59 34 49
- NSD – The Norwegian Centre for Research Data AS, by email: (personverntjenester@nsd.no) or by telephone: +47 55 58 21 17.

Consent form

I have received and understood information about the project *Configurable Well montage dashboard* and have been given the opportunity to ask questions. I give consent:

- ☐ to participate in a user-test session
- ☐ to that notes, video/audio recording may be taken during the session
- ☐ to the storage of my personal data. This includes video/audio recordings, written notes, work title/ name of company (where applicable), until the end of the master thesis, at the 30th of June 2019.

(Signed by participant, date)

G Test Information for the Different Prototypes

Master's Thesis Test Information 08.03.2019

Test Information for Paper Prototype

You will now be part of a user test for a new user interface. The user interface is created for oil industry, explicitly for Schlumbergers DrillPlan software. You will be testing an initial low-fi prototype that aims to test basic top-level functionalities. The prototype is by no means feature complete, and there will be situations that lacs visual feedback that you can interact with.

Intro to the product:
Well planning is typically a process that generates hundreds of pages of documentation and data. The application you are testing plan to help create a single paged summary of an entire well planning process. This in order for humans to get a usable overview. In our application, you can create a summary in a dynamic browser tool and there are some of these interactions we are testing today.

We are testing the user interface and how it communicates, not you. There are no wrong answers, and any feedback, questions or thoughts during the test are more than welcome.

How the test will be conducted:
We have created a test plan with a set of questions and tasks that you have to respond to. You will be testing two slightly different approaches today. We will be dividing these into Prototype 1 and Prototype 2. The tasks and questions will be similar for both of these approaches, if not entirely identical, but the interfaces will differ slightly.

The two different Prototype will be clearly announced and we may ask you to elaborate which you liked the most and why.

Lastly, we need you to think out loud before you do each action. This in order to describe your thoughts so we can better understand your mental model of the system.

Summary:

Intro to product and background
Prototype 1 User testing with set tasks
Prototype 2 User testing with set tasks
Questions for comparing and clarification
Comments from you

Remember to read and sign the "*Information Letter*" specifying your rights in accordance to Norwegian Centre for Research Data (NSD) laws and Norwegian University of Science and Technology (NTNU)'s rules for General Data Protection Regulation and Personal Data Act.

Page **1** of **1**