

# Interaction Design (INF2260/4060)

University of Oslo

Ouda Abuazra, Bilal Anwar, Michael Dejene Kifle and Jo Møller Warmedal

## Comfyguration

*”What function(s) can be added to Configuration Designer to improve its usability?”*



## Table of contents

<b>1. Introduction.....</b>	<b>3</b>
1.1 Goals.....	3
1.2 User group.....	3
1.3 Scope.....	3
1.4 Methods.....	3
<b>2. Configuration Designer.....</b>	<b>4</b>
2.1 Overview.....	4
2.2 Preliminary meetings with Schlumberger.....	5
<b>3. Data gathering.....</b>	<b>6</b>
3.1 Survey.....	6
3.1.1 Survey answers.....	7
<b>3.2 Interview.....</b>	<b>7</b>
3.2.1 Interview answers.....	8
3.3 Focus group.....	8
3.4 Ethical discussion about the data gathering.....	9
<b>4. Analysis.....</b>	<b>9</b>
4.1 Preparations.....	9
4.2 Analysis of the quantitative data.....	9
4.3 Analysis of the qualitative data.....	10
4.3.1 Open coding, conceptualization and categorization .....	10
<b>5. Idea proposals.....</b>	<b>12</b>
5.1 Low-fidelity prototype #1.....	12
5.2 Low-fidelity prototype #2.....	14
5.3 Idea feedback.....	14
5.4 High-fidelity prototype.....	15
<b>6. Usability testing.....</b>	<b>15</b>
6.1 Execution.....	16
6.2 The test data.....	16
6.3 Evaluation of the test data.....	17
6.4 Validity, reliability and issues.....	18
<b>7. Conclusion.....</b>	<b>19</b>
<b>8. Reference list.....</b>	<b>19</b>

## 1. Introduction

We are four students from the Institute of informatics at the University of Oslo. During the fall of 2016 we conducted a research project on behalf of Schlumberger, in context of the course INF2260: Interaction Design.

Schlumberger is the world's leading company in petroleum services, offering human resources, state of the art technology and training. Employees of the company working within the upstream sector of the oil industry often use Petrel – an in-house software application offering its users a wide array of services, tools and functions relating to upstream tasks. An in-house plug in tool called Configuration Designer can further on be used in conjunction with Petrel, to configure its GUI to the users liking.

### 1.1 Goal(s)

The goal of this project as requested by our client is to assess “What function(s) can be added to Configuration Designer to improve its usability?” by researching its users, the tool and how it is used.

### 1.2 User group

Our targeted user group is the approximated 600 users of Configuration Designer at Schlumberger.

### 1.3 Scope

From early on, there was little to no equivocation about what Schlumberger wanted from us and what we understood to be our task. Our main objective was never to generalize our findings, but rather to improve an already existing interface. We have therefore confined our project within the parameters of Schlumberger and our aforementioned user group.

### 1.4 Methods used

Our choices of methods were in large part dictated by the scope of our project. For the data gathering, we chose a triangulation of surveys, interviews, observations and a focus group, to map out a better understanding of the tool, the users and the issues they might have had with it.

Seeing as we were left with mostly qualitative data, we analyzed the data through open coding, conceptualization and categorization of the data. We furthermore reviewed some of the data with an expert at Schlumberger, who helped us narrow down further work by discussing practicality and the possibilities of implementation. We eventually created a high-fidelity prototype, before conducting a user based usability test to assess our design.

## 2. Configuration Designer

To have a formal understanding of our study, one must understand Configuration Designer and how it is used. As previously mentioned, CD (configuration designer) lets users of Petrel configure its GUI, but it is important to note the level of complexity it offers. The configuring of the GUI is almost exclusively confined to creating buttons with specified functions, and the arrangement of the buttons, primarily in the Ribbon of Petrel or the toolbar.

### 2.1 Overview of Configuration Designer

CD is divided into three panels. The left panel holds the configurations, the middle panel is where the user configure the elements (buttons and the such) inside a configuration and the right panel gives an overview of the already existing configured elements in the program. To create a button, the user can assign commands (lines of codes) to a 16x16 or 32x32 bit icon, give the button a caption and a description and decide which tabs and tab groups the button is in. One can program the functionality of a button oneself, or download plug ins created by system developers. As mentioned before, buttons (or the general term of configured elements) are made inside a configuration. Configurations help organize the work. An analogy that might be befitting, is that a configuration is like a windows folder (documents, images etc). CD comes with a set of predefined configurations, but the users can also create ones themselves. Predefined configurations can also have sub-folders, such as “Ribbon” or “Toolbar”. If one for instance creates a button under the ribbon sub-folder of the predefined configuration Geology-Ribbon, the button will be placed in the ribbon of Petrel when a Geology project is open. The newest configurations created are sorted by ascending date.

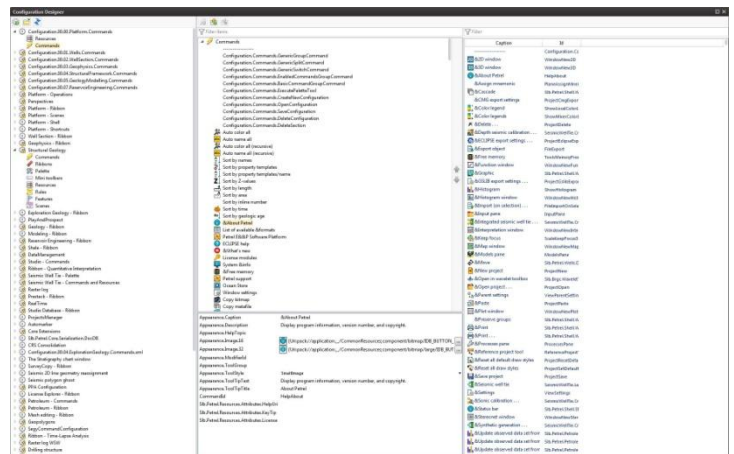
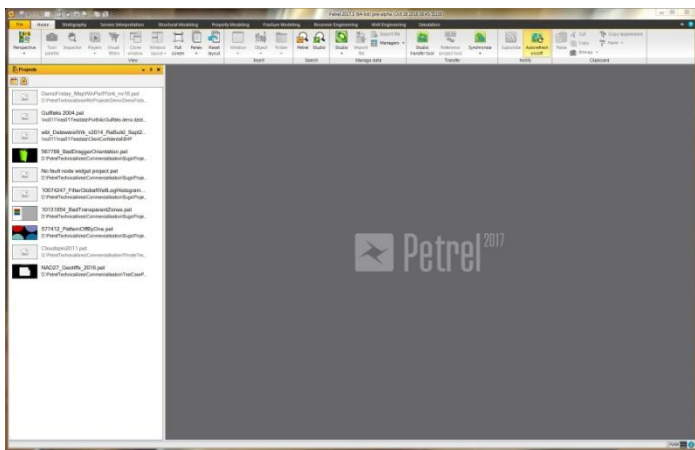


Image 1: Petrel and Configuration Designer (left and right respectively)

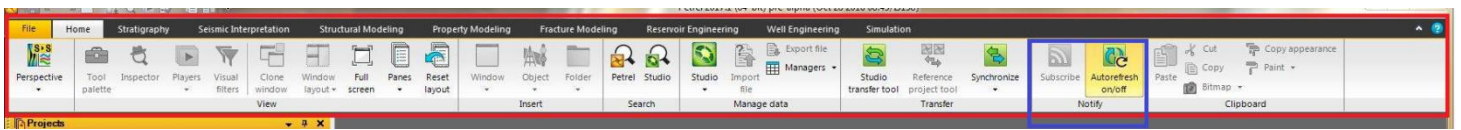


Image 2: The ribbon of Petrel (highlighted in red). Buttons are organized inside tab groups (highlighted in blue)

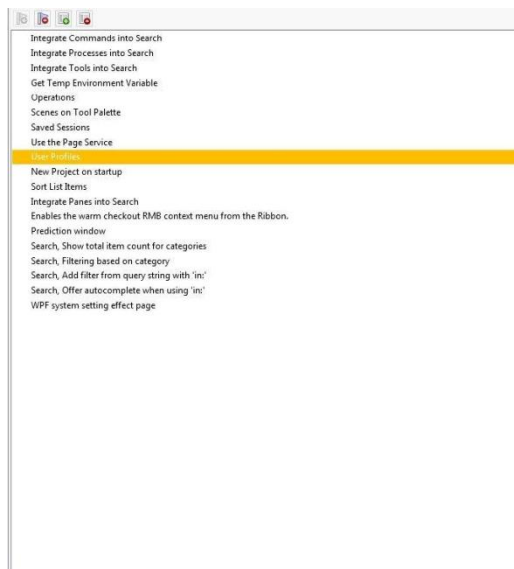
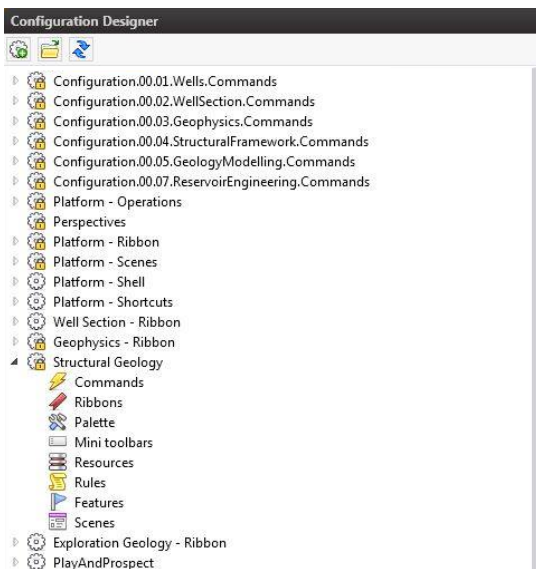


Image 3 and 4: Cliparts of the left and middle panel of Configuration Designer.

## 2.2 Preliminary meetings with Schlumberger

We had our first meeting with our Schlumberger representative (Robert Messenger) and a colleague of his on the 13.09.2016. The purpose of this meeting was for us to

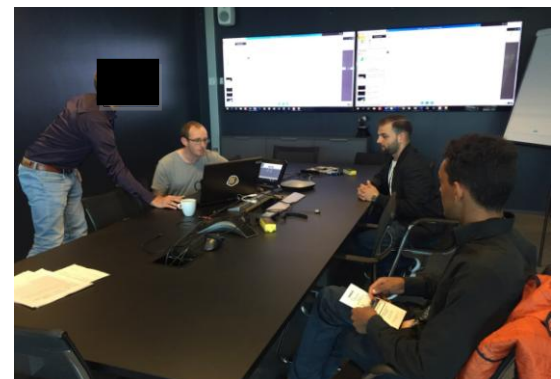


Image 5: Our first meeting. From left to right: Robert's Colleague, Robert, Bilal and Michael

introduce ourselves, and get a general overview of the project. Initially, we actually had the opportunity of choosing between this task and a second one about the iconography of Configuration Designer. Although the group had already discussed our preferred proposal before the meeting, we nevertheless requested a further brief on the both of them, on the oft chance we might change our minds. We went with the first proposal, as it was the preference of both parties.

To assess the usability of the tool, we sought insight from the users of CD at Schlumberger. Although we had not agreed on what data gathering methods we would employ at this point, the plan was to initialize that part of the project the following week. Despite the first meeting, we felt that we were lacking understanding of the tool. As of such, we were not confident enough to go through with the preparations for the data gathering process, seeing as it would have been difficult to inquire the users without really understanding the topic ourselves. We could not download CD on our private computers either, so we requested an additional meeting with Robert, inquiring on whether he could show us how the tool is used. He obliged the following week, as we attentively observed while he demonstrated the intricacies of the tool.



Image 6: Our second meeting. From left to right: Jo, Michael, Robert and Ouda

### 3. The data gathering

Though differing definitions might exist, we set the parameters of our project by the ISO definition of usability, “The extent to which a product can be used by specified users to achieve specified goals with **effectiveness**, **efficiency**, and **satisfaction** in a specified context of use”. (Culén 2015, p.4) Our data gathering methods were thus set up to measure these three metrics.

#### 3.1 Survey

Though we were a week behind on schedule, we now felt that we had the constituents needed to start the data gathering. The first item on the agenda was to attempt getting a general



overview over how the employees at Schlumberger are using CD, and their opinion of it, especially in regards to efficiency, effectiveness and satisfaction. As stated in chapter 5 of *Research Methods In Human-Computer Interaction*, surveys are a very effective tool for such undertakings (Lazar, Feng & Hochheiser 2010, p.101), and thus we created one using the free online service; SurveyMonkey. Using Robert as a conduit for getting our surveys out had two significant advantages. First and foremost, reducing the risks for coverage errors – relevant participants not receiving the survey – as all Schlumberger employees are issued with a work-specific e-mail adresse, which is where we sent it. Secondly, one might increase the response rate by using a trusted member of a community (Lazar, Feng & Hochheiser 2010, p.119). To ensure the questions were correctly formulated and of relevance to the study, we sent the survey to Robert to review it alongside a colleague to get the input of experts. We fortunately received some points of contention, clearing them up as to avoid measurement errors at any cost.

### 3.1.1 Survey answers

We sent the survey out to the approximated 600 users of CD at Schlumberger, garnering 38 responses in the space of a week, totaling up to a response rate of 6,3% of the total population.

Though we now had a better understanding of CD, we made the survey quite general in nature, as we initially were only interested

in an overview of the users. The last question of the survey was a blank textbox, where the participants could optionally voice specific issues they might have had with CD. It later turned out to be quite relevant for the eventual analysis.

## 3.2 Interviews

To supplement depth to the breadth provided by the survey, we conducted three semi-structured interviews with some of the employees. While being briefed, Robert emphasized the different ways CD is being used by the different people, in particular those with a background in IT (mostly system developers that create plug-ins for regular users), and those without. In addition, people also have different levels of experience with the tool, as some use it on a day

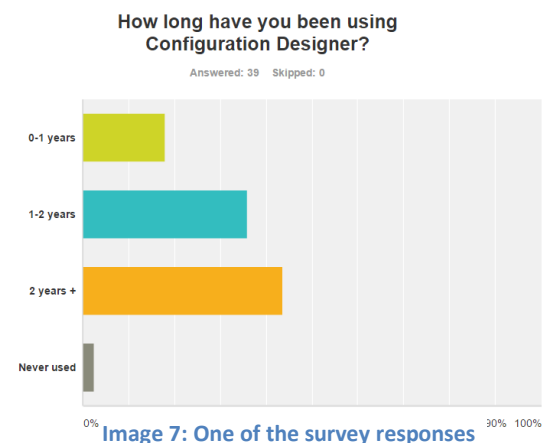


Image 7: One of the survey responses

to day basis, whereas others rarely do. With the help of Robert, we thus created three personas, system developers, experienced users and intermediate/novice users. With Robert himself falling into the middle category, we conducted a pilot interview with him, making some adjustments to the interview guide, before commencing with the interviews on the 28.09.2016.

### 3.2.2 Interview answers

As the date of the interviews approached, some of the participants unfortunately had to drop out. We were nevertheless able to perform three interviews, one with a system architect whom was involved in the development of CD, one with a not-so content end user with intermediate experience and one experienced end user. We performed semi-structured interviews as we expected the interviewees to touch upon things we would like them to elaborate on further. The interviews were set up so that one of us asked the questions, while another jotted down notes. The interviews were taped (with the participants consent), as to help ourselves further down the line with analysis. One requirement we had for the interviews was that the interviewees had access to their work-computers. The reasoning was that we understood how complicated CD is as a tool, and it would be both feasible and more efficient if they could demonstrate their workflows, instead of having to explain them step by step. This way we could also triangulate the interviews with observations. The interviews were performed at the offices of the participants at Schlumberger, to convene them.

### 3.3 Focus group

The reasoning for conducting our focus group was that we hoped the participants could feed off each other, and start meaningful and insightful conversations, with different viewpoints meeting. For the focus group, we employed two moderators to effectives the data gathering (Lazar, Feng & Hochheiser p.193), despite the fact that none of us were experienced with focus groups or HCI research from before. Likewise with the interviews, we chose participants with

Consent form	
<i>Description of the project</i>	
We're a group of four students from the Institute of informatics at the University of Oslo conducting a research on behalf of Schlumberger, and as a part of a course: INF2260. Our course leader is Alma Leora Culén, and our supervisor for the project is Dr. Hani Murad, both employed at UiO. The purpose of our project is to research the usability of Configuration Designer, and propose possible changes to the interface, hopefully making some improvements.	
We have gathered data from a wide variety of users through interviews, surveys and a focus group. Over a period of about a month, we've thoroughly analysed the data and constructed a high-fidelity prototype reminiscent of Configuration Designer's graphical interface. We've included some features users have explicitly asked for, and some that we've inferred to be useful through the analysis.	
We wish to conduct a usability test to gauge whether or not our proposed changes could help improve Configuration Designer. All participants will remain anonymous, and the data will only be used for further analysis, and exclusively by us. Participants will be asked to perform a couple of tasks individually, where we measure metrics such as time, and number of tries. The test will neither be video or audio recorded, but the participant will be asked to fill a small survey at the end, with the possibility of a few questions being asked. If the participant at any point wishes to leave, he/she will be permitted to do so without any scrutiny. Any data gathered at that point will be terminated at the request of the participant.	
.	
Consent: I've read and understood the above stated terms, and wish to give my full consent.	
_____	_____
Date	Signature

Image 8: The consent form given to the interviewees



different backgrounds and experience levels. The interviewees were excellent, and they participated evenly in the discussion, but there were nonetheless some issues. First and foremost volume. We only managed to have one focus group, which is often discouraged. Despite our efforts that was all we could muster up. The focus group also consisted of only four participants, though it is often advised to have five or more. Despite the apparent drawbacks, we still went on with it. We made sure not to rely wholeheartedly on the data, rather hoping to use it supplementary with the other data we had gathered if seen as reasonable. If the data were deemed as completely unreliable, we at the very least had the experience of conducting a focus group.

### 3.4 Ethical discussion about the data gathering and further work

Before each interview, we had the interviewees sign a consent form where we outlined the purpose of our research, how we intended to use the data and informed them of their rights, specifically their right to end the interview at any point without scrutiny and their right to anonymity. We taped the interviews and focus group, while also video recording the demonstrations (keeping their faces out of the frame), but only after the consent of the participants. All the participants remain nameless (including a domain expert whom assisted us later in the project).

## 4. Analysis

### 4.1 Preparations

The same day we finished the data gathering, we began some of the groundwork for our analysis by summarizing the interviews and gathering the data in a joint database (google drive) – to ensure a degree of validity to the data. The following day, the summarizations were done and we commenced with the analysis.

### 4.2 Analysis of the quantitative data

There were unfortunately some issues with the

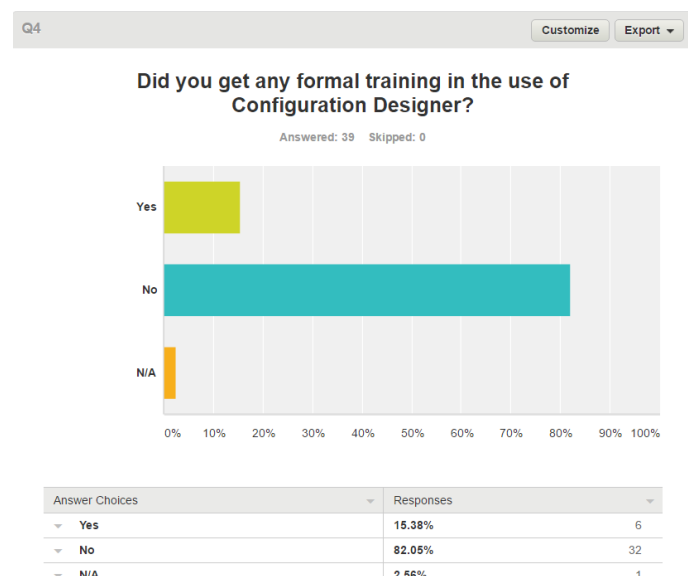


Image 9: The results from the referenced question

survey data. If we were to follow the standard confidence level of 95% (CheckMarket n.d), our sample size of 39 participants and total population of about 600 yielded a 16% margin of error. Using an example from the survey, one of our questions was “Did you get any formal training in the use of Configuration Designer?”. Approximately 80% answered that they did not, but with our rather large margin of error, the *true percentage* is somewhere between 64% and 96% ( $80 \pm 16$ ). Even if we lowered the level of confidence down to 90%, the margin of error was still a little too high for our liking (13 %). We thus opted to primarily use the text responses from the survey (which there were 20 of), and if some of the responses yielded rather large percentages, we would consider using that data supplementary (the aforementioned response of 80% would still be fairly large with the 16% margin of error subtracted, potentially indicating a significant trend).

### 4.3 Analysis of the qualitative data

We were then almost exclusively left with qualitative data, and thus opted for the first three steps of Grounded theory (open coding, development of concepts and categorization of concepts) as our main method of analysis. As previously mentioned, we had no intent of developing a theory through our study, and therefore assessed the last stage of Grounded Theory to be superfluous. The scope of the analysis was to find the issues people had with CD, and we were thus searching for concepts centered around that. Considering the specific nature of our study, we performed emergent coding (Lazar, Feng & Hoccheiser 2010, p.289), where we set out to interpret the data independently, before attempting to map out common threads.

#### 4.3.1 Open coding, conceptualization and categorization

The first bout of analysis lasted for about a week, with three meetings over the period. During our first meeting, we began the process, with three from the group given an interview to work with alongside all the text responses from the survey. Over the next two meetings, we individually searched for codes in the data, especially looking out for in-vivo codes (Lazar, Feng & Hochheiser 2010, p.291), as the interviewees were often of a technical predisposition (system developers, or just generally tech-savvy people). After we had finished our first round, we exchanged interviews and repeated the same process. This was done to ensure inter code

reliability, and also intra code

reliability by having each coder

perform the same task thrice. We

eventually compared our interpretations, first

making a joint list of codes, concepts and

categories which we refined after further discussion.

I would really like to be able to search for configured elements  
9/27/2016 1:29 PM [View respondent's answers](#)

As I am using it rarely, every time it is a struggle to identify, which element I need to modify, and where that element is in the configuration designer structure.  
9/28/2016 1:58 AM [View respondent's answers](#)

Image 10: Some of the textresponses from the survey

The most reoccurring issue people had (both from survey responses and interviews) was the difficulties finding both configurations, and elements configured inside a configuration. We believe this issue was formulated in two different manners. Either as an explicit complaint about the difficulties finding these elements, or as a wish for a type of searchbar (image 10).

Additionally, two out of the three interviewees mentioned a left panel searchbar in response to the question “can you think of any specific functionality you want that Configuration Designer doesn’t have?”. We categorized the above mentioned issues under the concept of “overwhelming”. During our initial bout of coding, all three coders had indentified this as a concept, though we might have been written it down differently (searchbar, search function, search). The intermediate user we interviewed mentioned “a lack of intuitiveness” as an issue with CD in general. He elaborated with how he constantly has to “relearn” essential aspects of CD after prolonged periods of not using the tool. This was further emphasized by some of the comments from the survey, such as “it’s simply not obvious how to use”. We borrowed the interviewees’ description of the problem as *in-vivo*, and labeled it under the concept of “unintuitive”. The single most explicitly requested function from the survey was a checkbox to enable or disable feature flags (4 out 20 responses). The way CD is at the moment, one enables or disables a feature flag by writing down a specific syntax in a text field (value//true or value//false). Though we attempted to avoid the data from the focus group, we could not ignore that three out of five participants also mentioned this as an issue, alongside Robert during the pilot interview. We filed this under the concept “complicated”. We were not only searching for the negatives of CD, as we wanted to make sure that we did not meddle with its positive aspects. One of the questions during the interview was “what are some of the

strengths of CD”, but twice we were not given an answer. We categorized all of the concepts under the categories of “efficiency issues”, “effectiveness issues” and “satisfaction issues” - and though it might have been difficult - “effectiveness positives” and “satisfaction positives” as we could not find anything to put under “efficiency positives”. The choices of categories was done so that we focused on the usability of CD above all else. Below is the end result.

**Efficiency issues:** Time consuming, overwhelming, unintuitive, steep

**Effectiveness issues:** Complicated, steep learning curve

**Satisfaction issues:** Messy, bad consistency, dull

**Effectiveness pros:** Unequivocal (means that after one has understood how a function works, it does what it is supposed to do)

**Satisfaction pros:** up to date

## 5 Idea proposals

After showing our findings and the responses from the survey to our to a domain expert (a system developer, who is also tasked with training people in using CD), we (alongside Robert) discussed the possibilities around implementation, and he further on relayed to us complaints some of his trainèes have had over the years. Considering the numerous issues people had with CD, we were aware that we could not cover all of them with an eventual prototype. Like many of the survey respondents, we believed the issues regarding the overwhelming nature of finding configurations and configured elements could be solved through a searchbar. Robert and the expert were quite positive to the idea, but there were some details that needed to be hashed out, namely how the data would be presented to the user.

### 5.1 Low-fidelity prototype #1

We thus created a storyboard of sorts, further elaborating the intricacies. In the left panel of the program, we proposed a searchbar with a filter, one for configurations, and one for configured elements. With photoshop, we made representations of how the search data will be presented.



Image 11: Clipart of the storyboard with the 'Configuration' filter active



Image 12: The configurations that matches the searchstring will be presented in the left panel

Below is a representation of how configured elements (buttons, tabs, tooltips and such created inside a configuration), will be presented if searched for (the configured element filter activated).

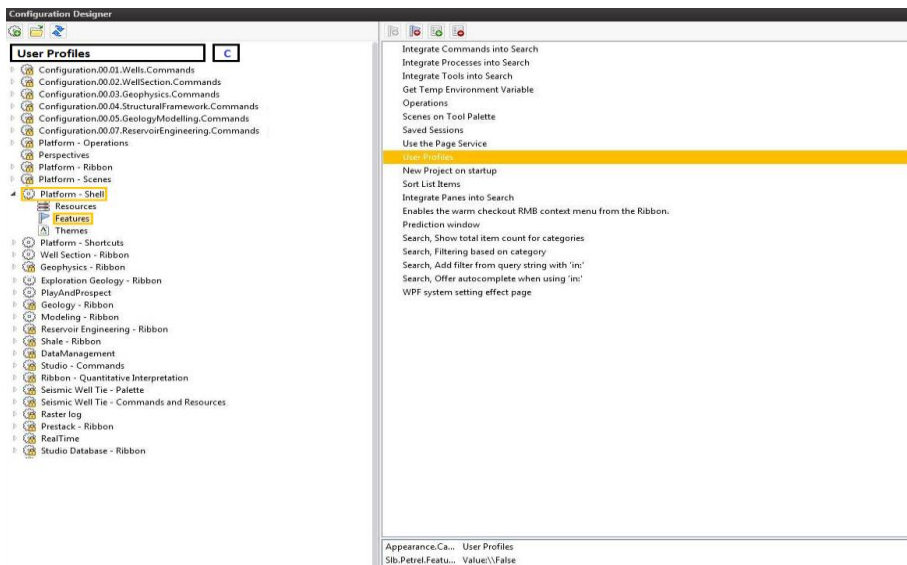


Image 13: How the searchresults are shown if the "configured elements" filter is active

As soon as a searchstring matches, the path to the configured element will be opened. That is done by highlighting the configured element in the middle panel, alongside the configuration

and the sub-category it is under in the left panel. Seeing as a personalized CD could have dozens of configurations, finding a configured element inside a configuration manually could be a very tedious task, and we believe that this would save the users significant amounts of time.

## 5.2 Low-fidelity prototype #2

Another function we believed could address some of these problems, would be a "help" tab in the program, where the users could get guidance from the program itself in some of its uses.

This could be done by employing some rather standard functions like 'tutorial' or 'FAQ' (frequently asked questions). We believed that

something such as this could address some of the issues with intuitiveness that reoccurred from the data. Referring back to a piece of information we gathered, 80% said they did not receive any formal training in the use of CD, despite it being offered at Schlumberger. Even subtracting our rather high margin of error, 64% is still a significant amount.

This coupled with the aforementioned issues of intuitiveness, thus made us believe that a function like this could be beneficial.

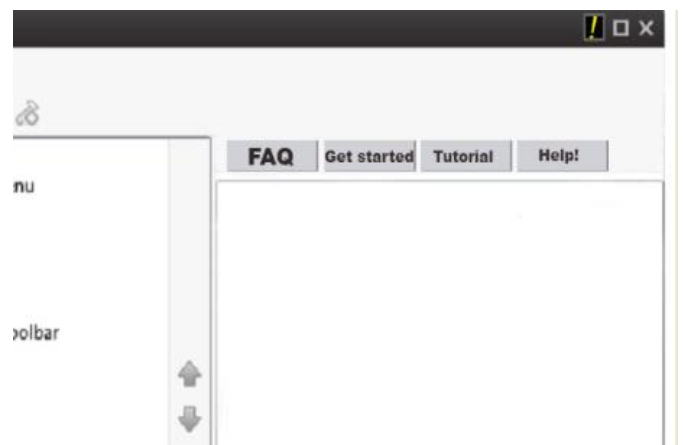


Image 14: A suggestion to how the GUI could look. The "help" elements placed in the right panel of Configuration Designer

## 5.3 Idea feedback

All in all, we believed that the first prototype could cover more of the issues, also doing so to a larger extent. Although a help tab might slacken the learning curve of CD, formal training is still offered at the company. We sought input from Schlumberger, and sent them our proposals. They reviewed them, and for our next meeting we discussed the pros and cons of each, while the expert helped assess the possibilities of implementation. Both parties agreed on the first proposal being the way to go while also wanting a way to enable feature flags with a checkbox. For the next 2 weeks, we worked on a testable, high-fidelity prototype of it, where we also addressed the issues with feature flags.



## 5.4 High-fidelity prototype

Programming in javascript and HTML, we made a “mockup” of CD with the proposed searchbar and six make-belief configurations with adjoining elements inside them. We used a screenshot of CD as the background, as to ensure comparability with the real program.

## 6 Usability testing

After finishing our high fidelity prototype, the next item on the agenda was to compile a large enough assembly to perform a user based

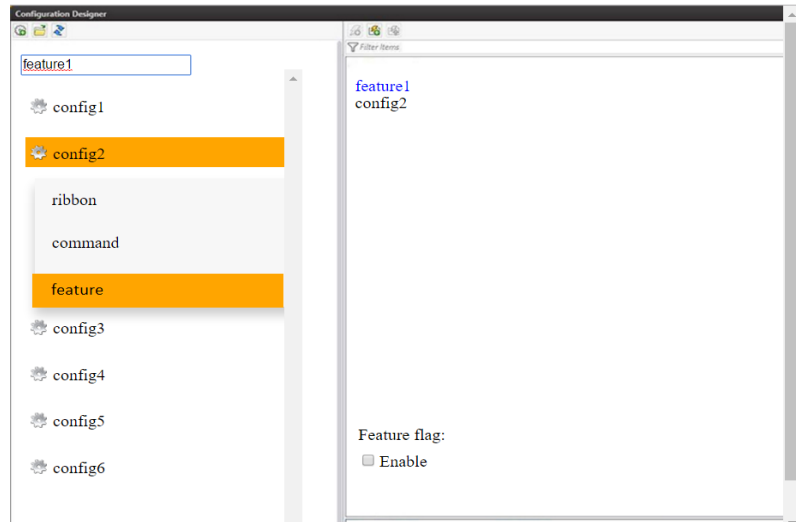


Image 15: Screenshot of the high-fidelity prototype of the configured element filter

usability test. In regards to usability testing, it is often said that 5 participants can uncover up to 85% of all usability issues (Nielsen 2000). This estimate has also been disputed over time. Two HCI researchers argued that websites they studied required a far greater assembly to uncover its usability issues, as the first 5 participants only found 35%. In the same article, it is also argued that the amount of participants needed is dependent on the interface that is being researched. The “5-person estimate” derived from studies done on the interfaces of the 90s, such as PC and mainframe applications (Bastien 2009, pp.18-24), and as we believe that CD holds many similarities to these, we aimed for something close to 5 people in the usability test.

Schlumberger provided us with the same people we had interviewed (eight people in total). That was of great value to the test, seeing as we had system developers and regular users alike, in addition the varying levels of experience amongst them. The test conducted at Schlumberger, and it was set up to measure the efficiency, effectiveness and satisfaction levels of our prototype compared to the real program. The users were given two similar tasks, were we recorded the time spent in both, before the subjects were shortly interviewed and given a questionnaire to fill after performing both tasks, to further triangulate the data gathering and getting qualitative data.

### Task 1

*"Can you find the feature (a configured element) "well model correlation" amongst a selection of 6 configurations in under 6 minutes?"*

### Task 2

*"Can you find the configured elements "feature 5" and "feature 6" in under six minutes, and enable their feature flags?"*

## 6.1 Execution

In the prototype, we had only made 6 configurations, and to make the sure both tasks were set up similarly so we could compare the results, we limited the amounts of configurations the users were to look through in task 1 to 6. So that the users would not forget which configurations were respectively the head and the tail, we wrote the names down on a piece of paper which they were given alongside the task description. Though our prototype was only a "mock up", we went to great lengths to ensure an almost identical GUI. With these details worked out, we felt that we had set up the right conditions to compare the times of both tasks.

We needed two computers for the test, one with the prototype and one with the main program. Robert was the first participant of the test, but seeing as we used his personalized CD, asking him to find a specified configured element would be counter intuitive. We thus used his test as a pilot, to see if everything was set up correctly.

## 6.2 The test data

As seen in the test table, only 6 out of the 7 participants managed to perform the first task within the time limit, yielding a **completion rate of 85.7%**, as compared to the **100% completion rate** of task 2. Furthermore, the participants averaged 00:40.6 minutes on the second task as opposed to the 1:58 minutes spent on the first, resulting in a **65.7% time reduction**.

Participants	Task completed?	Time (min)	comment
1	/	/	Didn't participate
2	N	Max	
3	Y	00:46	
4	Y	02:32	
5	Y	00:44	
6	Y	00:22	
7	Y	00:54	
8	Y	02:31	

Participants	Task completed?	Time (min)	comment
1	Y	1:59	Just a test round
2	Y	1:21	
3	Y	00:40	
4	Y	00:37	
5	Y	00:29	
6	Y	00:28	
7	Y	00:33	
8	Y	00:36	

Table code : Y=yes ,N=no . Max=6:min

We realized how it would be superfluous to ask them to rate the “time saving” aspect of the prototype, seeing as this was something we could calculate ourselves. We nevertheless gathered the remaining scores, and compiled the averages.

**Simplicity:** 4,1/5

**Usefulness:** 4,4/5

**Satisfaction:** 3,8/5

We got six very valuable text responses to the question “are there any additional features you would like to see added?”. Two of the responses revolved around “wildcard searches”, meaning that the searchbar presents matching data as the text is being written. Another respondent took issue to how an implementation of our proposal, would effectively lead to two searchbars in CD. There is already one in the right panel, but it only lets the users search through commands without showing where they can be found. He suggested that we merged both of them into one searchbar, as to reduce confusion. The responses to the new feature flag function quite positive across the board. After finishing the data analysis, we made a test report which we gave to our Schlumberger representatives.

### 6.3 Evaluation of the test data

We were aware of the presence of the Hawthorne Effect, as all the participants were carefully observed throughout the test. We nevertheless assessed its influence on the test data to be minimal, as all the participants were already highly skilled in the use of CD, which to a large extent also trickles down to their skill levels in the use of the prototype as it is a “mock up” of the real program.

Efficiency in the context of usability is defined as “How fast we can solve a task”(Joshi 2015, p.24). We set up the test to measure the difference in time performance between task 1 and 2, which as stated above was a 65.7% improvement in the second task.

Effectiveness on the other hand is defined as “How well we can solve a task“ (Joshi 2015, p.24). An improvement in effectiveness in this case would be an uptick in task performance. That was the case with a 12.5% improvement from task 1 to task 2. We also believe less time spent on the second task also indicates better effectiveness, as spending a significant amount of time manually looking for a configuration, detracts from a real task the user would set out to do (making a button for instance).

Lastly, satisfaction is defined as “How satisfied we are with solving a task“(Joshi 2015, p.24). Though the average satisfaction score from the questionnaire was relatively high at 3.8 out of 5, we do not believe we have done a sufficient enough job at measuring this metric (will be discussed in the next part).

#### **6.4 Validity, reliability and issues**

We have assessed the validity of the data to be rather moderate. We had set out clear goals for what we wanted to measure, but there were issues with some of the execution which we believe compromised the overall validity. Though we felt that efficiency and effectiveness were measured well, we unfortunately did not have good enough data to make conclusive statements about the disparities in user satisfaction. One improvement we could have made was to give the participants two questionnaires, one for CD and one for the prototype, before comparing the average satisfaction scores. Despite these issues with execution, we believe that triangulating observation, with short interviews and questionnaires helped strengthen validity.

In aftermath, we believe it would be better to have at least two bouts of user based testing. One reserved to CD alone, and another for the prototype. Our reasoning is that we initially could have mapped out the flaws of CD (supplementary to the data we already had gathered), worked on the prototype, and then performed a usability test with it in mind where we uncovered its flaws. As the gist of a usability test is to “improve the quality of an interface by

finding flaws in it” (Culén 2015, p.10), we believe it would have been more purposeful to separate the tests. We compared interfaces, but with more foresight, we might have employed a more befitting method for such undertakings. Something akin to experimental design perhaps, where we set out to disprove a null hypotheses such as “there is no difference in the time spent finding configuration x in Configuration Designer and this prototype”, and structuring the experiment accordingly to the research conditions. We believe that this mishap in choice of methods affected the reliability of the data from the usability test negatively.

## 7. Conclusion

We set out to research Configuration Designer and its users, attempting to uncover what functions could be added to improve its usability. Our choices of research and evaluation methods were dictated by this focus, as we conducted a rather concentrated study where we confined it to Schlumberger, its users and Configuration Designer, rather than conducting a broader study with a purpose of generalizing our findings. Throughout the project we were able to obtain beneficiary experiences in the use of several HCI-related research methods, not to mention knowledge. There were nevertheless problems with some of the execution of certain methods, and as of such we were not confident enough to arrive at unequivocally conclusive answers to the research question that permeated our study.

## 8. References

- Bastien, J.M.Christian. (2009). ‘Usability Testing: A Review Of Some Methodological And Technical Aspects Of The Method’. International Journal of Medical Informatics 79.4. Available from: <http://www.sciencedirect.com/science/article/pii/S1386505608002098> [28 November. 2016]. (Must be accesses while connected to the network of an academic institution)**
- CheckMarket. (N.D.).How to Determine Population and Survey Sample Size. [Internet] Available from: <https://www.checkmarket.com/blog/how-to-estimate-your-population-and-survey-sample-size/> [28 November. 2016].**
- Culén, A.L. (2016). ‘Week 10: Usability’. INF2260 – Interaksjonsdesign. Available from: [http://www.uio.no/studier/emner/matnat/ifi/INF2260/h16/chapter10\\_host2016.pdf](http://www.uio.no/studier/emner/matnat/ifi/INF2260/h16/chapter10_host2016.pdf) [28 November. 2016].**

**Joshi, S.G. (2015). 'Week 11: Evaluering'. INF1500 – Introduksjon til design, bruk og interaksjon.**

**Available from:**

[http://www.uio.no/studier/emner/matnat/ifi/INF1500/h15/undervisningsmateriale/forelesning\\_151027.pdf](http://www.uio.no/studier/emner/matnat/ifi/INF1500/h15/undervisningsmateriale/forelesning_151027.pdf) [28 November. 2016].

**Lazar, J., Feng, J.H. and Hochheiser, H (2010). *Research Methods In Human-Computer Interaction*. 1st ed. Chichester, West Sussex, U.K.: Wiley. Print.**

**Nielsen, J. (2000). Why You Only Need To Test With 5 Users. Nielson Norman Group.[Internet]**

**Available from:**

<https://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/> [28 November. 2016]