# User Interface for Event Configuration

## - An experimental study

LTH School of Engineering at Campus Helsingborg
Department of Computer Science

Bachelor thesis:
Mehmet Micoogullari
Simon Lerpard

# Abstract

Many companies today have a security system containing of devices such as cameras and other devices with some trigger functionality built in. It's essential to be able to control all existing triggers in a security system for it to work flawlessly. As the number of triggers increases, the security system in question becomes more complicated to configure. Therefore, making configuration from an easy to understand interface for the user is of great importance.

The purpose of this thesis is to make a suggestion how to configure triggers and actions for one or multiple devices through an interface in a graphical user interface.

To draw a conclusion considering this, current AVHS was tested to understand how the existing system works and how to add functionality to it. A few interview sessions were held with people familiar with Axis Video Hosting System(AVHS) to get their views of what functionality they would like to add to the system and how they would like it to work. Literature studies has been made on designing interfaces as well as a design studio workshop were held with employees in Axis. The purpose of the workshop was to get an idea of in which order they would like the flow of configuration to be and how the interface should be designed.

Two interfaces were designed and created based on the interviews and the design studio. They were then tested using a number of metrics to find out if they would have the wanted result from the test users. The results from the test sessions were then analyzed and presented in the thesis.

Analysis of the results showed that user interface 1 were more effective in aspect of effectiveness and time efficiency while user interface 2 showed to be more satisfactory according to the test users. Even if user interface 2 showed to be more satisfactory, the difference were not considerably big. This lead to the conclusion of that user interface 1 is the more appropriate user interface for event configuration.

**Keywords**: AVHS, Configuration, Triggers, Actions, Relations, UX, Interfaces, Event Configuration

# Sammanfattning

Många av dagens företag har någon typ av säkerhetssystem installerade som består av kameror och andra enheter med triggerfunktionalitet inbyggd. Det är absolut nödvändigt att man kan styra hur alla triggers skall hanteras för att ha ett säkerhetssystem som fungerar felfritt. Samtidigt som triggerantalet ökar blir ett säkerhetssystem allt mer komplicerat att konfigurera. Därför är det viktigt att konfiguration kan ske från ett gränssnitt som är överskådligt för användaren. Syftet med examensarbetet är att bidra med ett förslag på hur konfiguration av triggers och actions för en eller fler enheter med hjälp av relationer kan hanteras i ett grafiskt användargränssnitt.

För att komma till en slutsats angående detta så testades det nuvarande Axis Video Hosting System(AVHS) för att förstå hur det existerande systemet fungerar och vad som kan läggas till i det. Ett par intervjusessioner hölls med personer anställda på Axis som är insatta i AVHS för att få deras synpunkter på vad de tyckte saknades i systemet och hur de skulle vilja att det ska fungera. Undersökningar gjordes på hur man designar användargränssnitt och en design studio workshop genomfördes med anställda på Axis. Syftet med workshoppen var att få en idé om i vilken ordningsföljd de skulle vilja att konfigurationen skulle ske och hur designen på användargränssnittet skulle vara. Två användargränssnitt designades och skapades med intervjuerna och workshoppen som grund. De testades sedan med hjälp av utvalda metriker för att undersöka om förväntat resultat skulle uppnås av testanvändarna. Resultaten från testsessionerna analyserades och presenterades därefter i examensarbetet.

Analys av resultaten visade att användargränssnitt 1 var mer effektivt i aspekt av effektivitet och tidseffektivitet medan användargränssnitt 2 visade sig vara mer tillfredsställande enligt testanvändarna. Även om användargränssnitt 2 visade sig vara mer tillfredsställande var skillnaden inte märkvärdigt stor. Detta ledde till slutsatsen om att användargränssnitt 1 är det gränssnitt som är mer lämpligt för event konfiguration.

**Nyckelord**: AVHS, Konfiguration, Triggers, Actions, Relationer, UX, Användargränssnitt, Event Konfiguration

# Acknowledgements

# Contents

# List of Figures

# Terminology

In this chapter, a set of important words and acronyms are described. The presented words and acronyms are used frequently in this bachelor's thesis.

**Action**  - What will be performed when a trigger is fired

**AVHS**  - Axis Video Hosting System

**Batch**  - A selection of a group

**Event**  - The configuration flow from when something is triggered to an action is taken

**I/O box**  - A device used for connecting an Input/Output signal

**Logical operators**  - A set of characters which is used to create connections between data

**LTH**  - Faculty of Engineering, LTH, Lund University

**VMD**  - Video Motion detection. A sensor with the purpose of detecting movements in a video feed

**Mockup**  - A sample of an interface

**PIR**  - Passive InfraRed sensor. A sensor with the purpose of detecting movements, works in dark areas as well

**Relation**  - A connection between devices, triggers and/or actions

**Site**  - An area with a set up security system

**SUS**  - System Usability Scale

**Trigger**  - A condition that must be fulfilled for something to happen

**UX**  - User experience

# Chapter 1

# Introduction

This chapter contains the background and purpose of the thesis and describes Axis, the company with which the authors have collaborated. It also contains the questions at issue and limitations of this thesis work.

## 1.1 Background and Purpose

This thesis was performed in collaboration with Axis Communications hereby referred to as Axis. Axis is a company with the main goal of developing network devices such as cameras, microphones and speakers. The company has different kinds of cameras at their disposal to connect to any of their available security systems. The devices can thereafter be adjusted according to the configurations in their cloud service, Axis Video Hosting System, hereby referred to as AVHS.

What Axis wants is an interface with the possibility to build relations between triggers and actions. The meaning of that is when something triggers, more than one device can take one or many actions.

An example of this would be *to set up two cameras, camera A and camera B. They are configured to trigger on when they detect any movement, which will lead to the two cameras taking an action. The action taken will be that both camera A and camera B will start recording. In this given example, when someone walks into the field of view of any of these cameras, they would both start to record simultaneously.*

## 1.2 Objectives

The objectives of this thesis is to develop a set of interfaces which will be used for event configuration. These interfaces will make it possible to configure relations between triggers, actions and devices for a chosen schedule.

The interfaces will be developed and thereafter compared to each other by usability tests with the help of prepared use cases. They will then be evaluated with predefined metrics and observations. Notes will also be taken during the tests. By analyzing the data gathered from the usability tests, an interface will be determined as the most suitable one for the purpose of event configuration.

## 1.3 Questions at Issue

To be able to reach the goal for this thesis, some questions needs to be answered. The questions are focused on how use cases will be elicited and what measurements needs to be taken into account to find which of the proposed interfaces that handles the use cases best.

- How will the use cases be elicited?

- Which interface is the most time efficient?

- Which interface is compatible to most use cases?

- How will the configuration of events be saved?

## 1.4 Limitations

The scope for this thesis is to analyze which of the proposed interfaces is the most suitable one for event configuration depending on user input and compatibility of the use cases. The interfaces are suitable for two to six devices but should support configuration of up to 16 devices. Systems in need of more than 16 devices are not suited for AVHS. This thesis does not contain any analysis of other desired features i.e how to generate a standardized installation report. This is described more in Chapter 5.4.

## 1.5 Thesis Motivations

Axis had a lot of thesis ads available to apply for. The ad for this thesis seemed very interesting but was too wide and needed to be specified a bit further. After some communication with Axis, the ad became more interesting after being remarkably more specified and the scope of this thesis was decided.

The subject of the ad, and later on the scope, was in line with the education, computer science. Containing elicitation of requirements, investigations, graphical and functional software development and also included some testing.

## 1.6 Thesis Outline

**Introduction** This chapter contains the background and purpose of the thesis and describes Axis, the company with which the authors have collaborated. It also contains the questions at issue and limitations of this thesis work.

**Technical Background**  This chapter describes Axis and their clients. The technical background refers to how AVHS works, which devices they have to their disposal as well as which customers AVHS is suited for. The event configuration of triggers and actions in AVHS today is explained and compared to how the employees wants it to work. Some features employees of Axis wants to see implemented and further developed into the current AVHS in the future are also included in this chapter.

**Methodology**  This chapter describes the process of how the methods are used in the approach section. It describes the methods used in this thesis and explains for which purpose they were chosen. Furthermore, it explains the purpose and execution of the design studio as well as how the interfaces will be usability tested and the tools needed for the results to be measured.

**Result**  This chapter contains the results of the methods used during the thesis. It also contains the analysis of the design studio workshop as well as the interfaces and the usability tests on them. This chapter also contains a section where the presented results are discussed.

**Conclusions**  This chapter presents the conclusion of the whole thesis. The conclusion of the questions at issue, as well as the result of the final interfaces are presented and discussed in this chapter.

**Future Work**  In this chapter, features or functions which were desired from Axis side, but not within the range of the scope for the thesis are presented in a list. The list is a suggestion to what might be of importance for future work.

# Chapter 2

# Technical Background

This chapter describes Axis and their clients. The technical background refers to how AVHS works, which devices they have to their disposal as well as for which type of customer AVHS is suited. The event configuration of triggers and actions in AVHS today is explained and compared to how the Axis stakeholders want it to work. Some features employees of Axis want to see implemented and further developed into the current AVHS in the future are also included in this chapter.

## 2.1   Axis Communications

Axis Communications AB is a Swedish-based company that acts globally via its own offices. Axis have been working with video surveillance since 1996. There are network cameras developed by Axis installed in public places such as airports, universities, prisons, banks, casinos and trains. At the end of 2015, the company had 2,139 employees and a revenue of 6.635 billion Swedish crowns (SEK). [2]

Axis was founded 1984 by two Swedish students, Martin Gren and Mikael Karlsson. They started off in the printer sector and continued on to become the world leader in network video. [3] Axis handles event configuration for all their devices with an interface in AVHS. The configuration available for events as of today is enough for Axis but they would like to add functionality to the event configuration interface in AVHS.

The devices in an event can currently only be configured individually according to a small set of triggers and actions. The option to create a relation between the chosen device and other devices is very limited. This means that cooperation between devices is not recommended since it is very fragile and the relation between them can easily be broken. Being able to configure these kinds of relations are essential as the customers wants to be able to use this feature.

## 2.2   Axis Video Hosting System

Axis Video Hosting System (AVHS) is a cloud service where the user can view and configure Axis network cameras. If there are recorded videos, they can be accessed online as well as downloaded from AVHS. It is possible to handle up to 16 devices in AVHS per site, which makes the service optimal for small to middle sized business companies which are in need of two to six devices in their facilities. However, the clients of Axis' AVHS are mainly alarm companies.

Figure 2.1 shows how AVHS is used today. The Axis hosting provider customizes a portal interface to meet the video service provider's graphical identity. The end user only sees the logo and name of the service provider. They can also watch live video by logging in to the service provider's video service portal. It can be watched from multiple devices such as computers and phones.
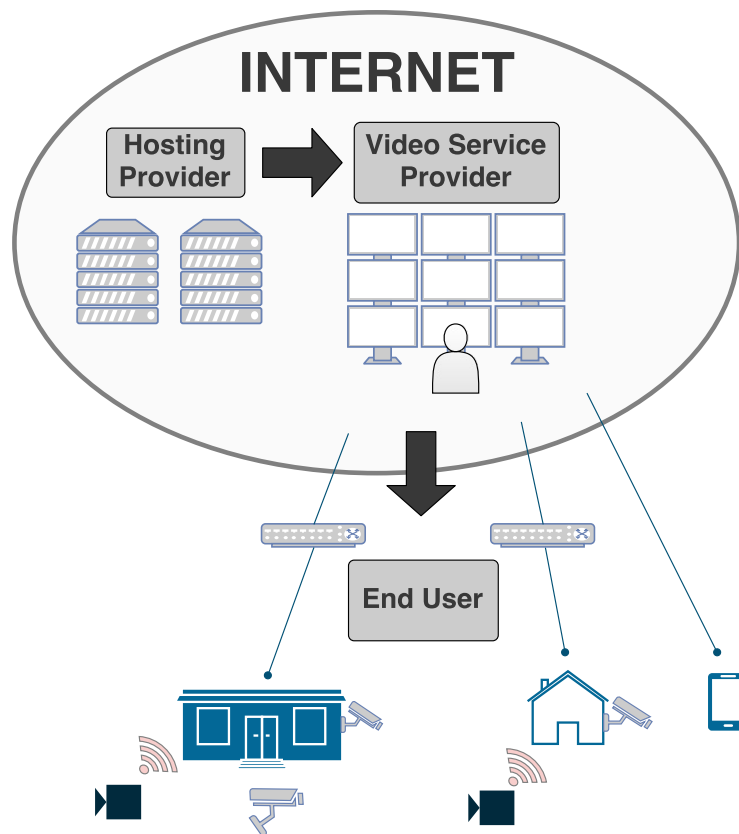


**Figure 2.1:** Shows how AVHS is used today.

### 2.2.1   Current Configuration Possibilities

The current AVHS configuration interface has a lot of features available, where some might be harder to configure than others. The configuration process may be difficult to understand in the beginning because of the setup structure and the need of pre-configured settings for specific tasks.

For example if the task is to setup a configuration for a camera to send a notification

when movement is detected, these steps need to be taken:

1. Add a recipient.

2. Navigate to an action rule overview for the camera.

3. Add a new rule to fulfill the trigger condition and select the action to send a notification to the added recipient.

If the recipient isn't added first, it can't be chosen during the action rule setup. And it's not possible to add a recipient during the action rule setup without aborting the process and restart the configuration for that specific rule.

As mentioned before there are a lot of features available. Another example of one of the features is the possibility to configure actions for events triggered from another device, by using a feature called HTTP trigger. An HTTP trigger is a feature where you can send a message from one device to another to tell the device to trigger by an IP-address. Using this type of trigger can lead to a very tricky and complex configuration even for a small surveillance system and it's not a reliable solution. The configuration could easily be broken by future updates or by the IP-address being changed. Figure 2.2 contains of a flowchart to describe how a rule for a device is configured.
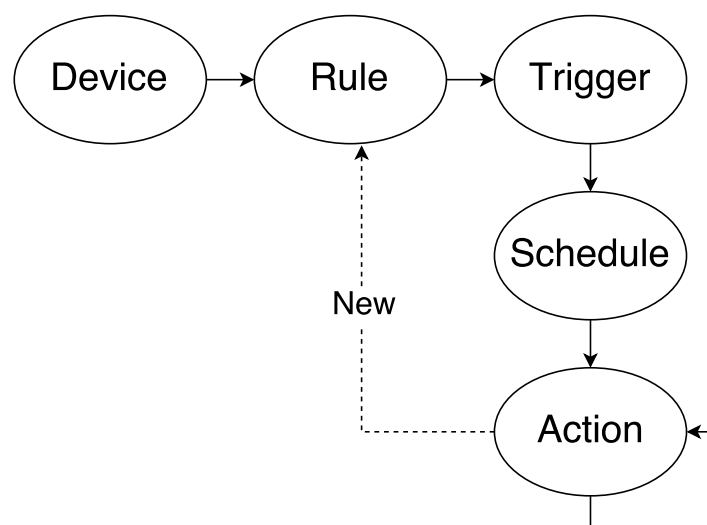
**Figure 2.2:** A flowchart for current AVHS interface

## 2.2.2 Desired Features

According to stakeholders from Axis, there are a number of desired features. There are also some features with room for improvement in AVHS. A list follows of the features and an explanation for each of them. The list is collected by analysis of the interviews in Appendix A.

**Batch settings** - Batch settings refers to the possibility of configuring multiple devices at the same time, in batch so to speak. A way of doing this could be to make the configuration once, save it in a template to be a so called pre-configured configuration.

It can then be used in other sites which require similar or the same configuration. Usually 50%-80% of all the configurations are made in the same way according to the participants of the interviews. Therefore it's important to be able to simplify the process as much as possible. A template or some kind of configuration in a batch would be desirable.

**Device replacements** - This feature would be very useful if a device breaks down in an already existing system. In the current way of replacing a broken device, the broken device would be removed and a new one installed in to the system, and thereafter reconfigured according to the settings the broken device had. A smooth way of replacing a broken device to *inherit* the settings of the one replaced would be a good solution.

**Installation report** - When an installer comes to a site to set up the devices ordered, there is no standardized procedure of how to create a report. A report about the installation process, which devices were installed, how long it took the installer to set up the system or which difficulties encountered and solved were on the wishlist. The main goal for the report was to summarize the process and to get a good overview on how the system is configured and installed.

**Multiple conditions** - This feature could be interpreted in different ways. Multiple conditions between triggers, actions or devices would give the configuration process further possibilities. An example of this could be that If condition #1 and condition #2 is fulfilled, one or more actions should be started. The relation between condition #1 and condition #2 is defined by using the two logical operators AND or OR.

**Wizard** - There are different kinds of ways to approach a configuration. One way to approach it could be as a wizard. A wizard works like a step by step guide to achieve a specific configuration. This way of configuration was requested by the participants of the interviews.

**Account security levels** - In the interviews, different security levels were described by the participants. A way to be able to control and configure a normal user account directly from an administrators account without the need of knowing the normal user login credentials were on the wish list. This could be achieved by having a higher security level like an administrator mode.

# Chapter 3

# Methodology

This chapter describes the process of how the methods are used in the approach section. It describes the methods used in this thesis and explains for which purpose it was chosen. Furthermore, it explains the purpose and execution of the design studio as well as how the interfaces will be usability tested and the tools needed for the results to be measured.

## 3.1 Approach

The approach of this thesis can be split up into three parts. The first part being gathering information, second being searching for requirements and the third one finding a solution by combining the first two parts.

The purpose of the information gathering was to make the objective of this thesis more visible and clear. In the beginning, meetings with relevant stakeholders were set up with the help of the mentors to gather the information needed to set a scope. With the scope and objective defined, the first phase had come to an end and the transition to phase two was made.

Two mockups (see Chapter 3.2.1) with different designs were developed in a program called Balsamiq with an initial idea of a solution in mind. With interviews being a good way of getting knowledge about the present work of a company, (see Chapter 3.2.2) a chosen few of Axis employees were handpicked and interviewed for requirement gathering.

During each interview, the roles of taking notes and asking questions were shuffled between the thesis workers. There were a total of four persons chosen for interviews. Every interview was performed in Swedish and started with asking the permission of the participant to audio record the interview. A questionnaire was made to be used during the interview with questions seen as relevant for the participant. This questionnaire was slightly modified for each person of the four. The developed mockups were printed and used as paper mockups to ask each participant for their opinions of the two printed mockups during the interviews. The aim of asking for their opinions were to get an idea of what

was appropriate in an event configuration according to the interviewed, and to get an idea of how future interfaces could be designed. The participant were free to ask questions and make statements of their own through the whole interview, as well as during the explanation of the two mockups. Every interview was summarized and translated from Swedish to English with the help of the recordings. The interviews helped the thesis workers to understand how the current system works and where it could be improved.

With that said, the gathering of requirements continued with a brainstorming session. The main focus of the brainstorming session was to find areas where security systems are important and use them to create use cases. A handful of scenarios were gathered and translated into use cases which would act as a requirement sheet for what the system would be able to configure. Furthermore, research were made on the steppingstones of how UX works and what was important to think of when designing interfaces. Books on interaction design[12], UX design methods[9] and user experience[17] were read to get a better understanding of the subject. From there on, a design studio workshop was organized with the main goal of finding out what the participants wanted to add to AVHS and how they wanted things to be configured. The design studio workshop also treated designing the interface to give the thesis workers some inspiration for future development. The thesis workers analyzed the gathered material and transcribed it to extract any useful information of what an interface could consist of and how it could be designed.

The requirements were set and the thesis workers continued on to the third step - the solution.

Two paper prototypes were made based on the data gathered from previous analysis. One of the prototypes was created with a graphical approach, which means that the interface was based on seeing much of what could be done in an advanced view, with drag and drop options. It was made in such a way that the user could decide to configure actions and triggers of devices in an order they were comfortable with. The user can configure triggers, actions or devices in which order they would like to. The other prototype was created as a wizard. The meaning of that is that a configuration is being built gradually, while the interface guides you trough each step. Each step has a text explaining what the user can do depending on the current tab the user is in. The two prototypes were then created as web-based applications, and were then tested for usability. Data was gathered by observations, taking notes and measuring some predefined metrics. By analysis of the data gathered from usability tests, the result of which interface was the most appropriate one for configuration of triggers were decided on.

## 3.2 Requirements

The following methods were chosen by the thesis workers to be able to set the requirements for the interfaces to be created. Each section describes the basis of a method and why it has been chosen.

### 3.2.1 Mockups

A mockup is a kind of presentation of a product's appearance and shows the basics of its functionality. Usually it is static and shows visual details. The viewer of a mockup

should get a good idea of how the final product will look, and a rough idea of how it might function, even if the functionality does not work. Mockups are often used to impress the stakeholders and investors and is a great way of generating interest.[6]

Mockups is a good way of getting feedback, critique and discussion of the design by usability testing them, which is one of the main reasons to why doing mockups were chosen. Other than getting feedback or critique, a mockup can also be used for teaching, demonstration, design evaluation or promotion.

Mockups can be created in a program called Balsamiq, which launched in June 2008. It is a graphical tool used to sketch out user interfaces for websites. These websites can be adjusted to fit desktops, mobile phones or mobile/web applications. [4]

## 3.2.2 Interviews

Interviews were chosen as one of the elicitation techniques because it's a good way of getting knowledge about the present work and the present problems. It can give information of opinions about what is realistic and where the conflicts may lie. By doing interviews, a large amount of information is gained and it usually gives a close insight of the current system and its missing features after being analyzed. Interviews also gives a lot of great opinions that could be used for future work. If there is a variation of the people interviewed, there is also a lot of different opinions and information from multiple perspectives that is useful for the elicitation of data. It's considered as the main elicitation technique of many analysts. [16]

There are four types of interviews: open-ended or unstructured, structured, semi-structured or group interviews. The first three types are named according to how much control the interviewer imposes on the conversation by following a predetermined set of questions.[12]

The format of the interviews used for this thesis were unstructured interviews. There was an agenda with some open questions prepared, with the purpose of having the interview to run more as a conversation than a questionnaire with the purpose of gathering as much information as possible over a short period of time.

All participants chosen for the interviews work at Axis in the department of New Business. The variety of roles among the participants differs greatly from software developers to product manager and UX advisor. This helps us to see the system from a wide perspective, with the input seen from different points of view from the positions in the company. They are all in contact with the AVHS service in some way or another. The participants work with it every day but with different parts of the system since their tasks differ. To show the difference in tasks between the roles, some examples of tasks will be given. The assignments could consist of creating feature boards, draw flowcharts, developing software, bug fixing, customer contact and demonstration of AVHS for potential customers.

## 3.2.3 Brainstorming

In a brainstorming session, a group of people are gathered to come up with ideas without the risk of being ridiculed in a focused atmosphere. All the ideas are being noted on a whiteboard, which leads to each idea spawning new ideas. While most ideas might be ordinary, the group might come across some very promising ideas. An important rule of

this method is to not criticize any idea. As a result of not criticizing the ideas generated, there will be many unrealistic ideas but even the seemingly stupid ideas may turn out to be of value later on. [16]

A brainstorming session was held to think of scenarios where security systems could be needed, and for what purpose. By doing the brainstorming session, finding out which triggers and actions could be utilized in reality were decided. When the triggers and actions were decided it was not hard to make up a scenario, and make it in to a use case where the relation between triggers and actions are explained.

## 3.2.4 Use Cases

A use case is a description of set of sequences or actions to achieve a goal. By doing a use case analysis, usage requirements for a new software program or task can be completed. Use case analysis is an important and useful technique not only for the analysis phase, but also in the design phase or in the implementation phase. The advantage of writing use cases are that basic requirements can be found. The use cases are only interesting when they are relevant. The use cases often describe how a user manages the system by listing the the steps of how the actions are being made.[16]

## 3.2.5 Design Studio Workshop

Design Studio is a creative way of approaching a solution for a specific problem. This is done by together as a team coming up with a solid suggestion of how to solve the specific problem. By focusing on a specific format as a group, a shared understanding of the problem can be created. There are five specific steps to take for this method to reach a conclusion as follows. According to Martin Christensen who is the creator of the Modern UX website, Design Studio is a good way of gathering new ideas with the help of collaboration. [7]

The design studio procedure are visualized in figure 3.1.

**Illuminate** - The first step is to get all the members of the workshop on the same boat by giving them a presentation of what the problem is, as well as the constraints. The constraints can be that a system has a specific target group or should be run on a certain platform.

**Sketch** - The second step is the creativity part. Everyone in the team will sketch solutions to the problem within a specific timebox, for example five minutes. The time has to be short so that the sketching is quick and dirty, since giving too much time to the group might lead to them getting stuck on unnecessary details.

**Present** - The third step is about everyone presenting the design each member made. The timebox of the presentation should be short, around one minute. When a member has presented their solution, a critique session for the person's design follows.

**Critique** - The fourth step is the critique session, which is an open discussion about the design last presented. The critique session should be short with one main question in focus, does the design solve the problem? The purpose of the discussion session

is to inspire the creator of the design to new ideas or how to further develop the presented solution by highlighting the flaws and the key issues of the design. In this step, every member of the workshop will think deeper about both the problem and the solution to come up with improvements of their own design for the next sketching iteration.

**Iterate** - The fifth step is to run the last three steps at least 2-4 times since the key to finding reliable solutions is iteration. It is important to not make too many iterations since it can lead to the group running out of ideas and the presented solutions might then not improve. By iterating, the team also gets a shared understanding of the problem.

The overall rule for design studio is to not dwell on details and try to get most value out of the least amount of time. After the design studio session comes to an end, the creator of the workshop has plenty of material to work with to make their design go forward, towards an implementation. [7]

```
┌─────────────┐
│  Illuminate │
└─────────────┘
       │
       ▽
┌─────────────┐
│   Sketch    │◁──────┐
│   (5 min)   │       │
└─────────────┘       │
       │              │
       ▽           Iterate
┌─────────────┐       │
│   Present   │       │
│   (1 min)   │       │
└─────────────┘       │
       │              │
       ▽              │
┌─────────────┐       │
│  Critique   │───────┘
│   (2 min)   │
└─────────────┘
```
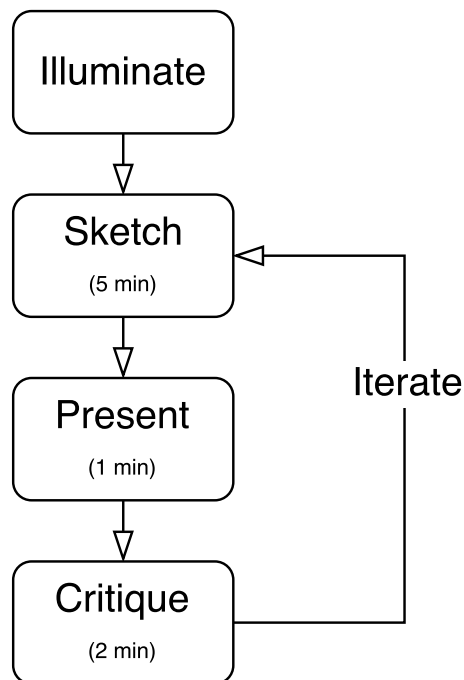
**Figure 3.1:** Design studio workshop structure.

## 3.3   User Interfaces

A user interface makes it possible for the user to configure settings for a device by communicating with the hardware.

When designing an interface, thinking of conceptual designs can be a useful tool. A conceptual design is a way of thinking of something that works in a certain way, and then design an interface in the same way it works. This makes the conceptual design used to work like a metaphor.

An example of this could be a map over a railway. Translated into design, this would mean that the user gets to see an overview of all pages on a website, and which ways needs to be taken to access them.

A design studio was set up to confirm the studied methods on how interfaces can be designed. Two new mockups was made to summarize and analyze the result of the workshop. These mockups would then be used when creating the interfaces for the usability test.

## 3.3.1   About the Interfaces

An interactive user interface can be a difficult and complex to develop. There are tons of languages and environments to choose from. The following languages, frameworks and libraries are used for the two interfaces developed as prototypes to use in the usability test.

**HTML -** Hyper Text Markup Language (HTML) is a markup language designed to describe a web page using markup. Web browsers reads HTML and renders the content of the page. HTML contains of different types of element tags. [23]

**CSS -** Cascading Style Sheets (CSS) is a language used to describe the style such as colors, images, borders, margins and other layouts of a HTML document and its elements. [22]

**JavaScript -** JavaScript (JS) is one of the three main languages used for the web (HTML, CSS and JS). It's a programming language for the HTML and the web. It was invented in 1995 by Brendan Eich and became a ECMA standard (ECMA-262) in 1997. Mozilla foundation is currently developing JavaScript.[24] JavaScript is usually supported by all major browsers and it's typically used as a client-side script language[19].

**jQuery -** jQuery is a JavaScript library used to traverse, manipulate, handle events, create animations on HTML documents. It's a fast, simple and feature-rich library that has changed the way millions of people write JavaScript because of a combination of versatility and extensibility.[13]

**Bootstrap -** Bootstrap is one of the most popular open-source front-end frameworks. It was originally developed by a developer and a designer at Twitter. The first release was released August 19, 2011 and has been rewritten two times since then (v2 and v3).[5]

**PHP -** PHP (recursive acronym for PHP: Hypertext Preprocessor) is an open-source server-side programming script language suited for web development. PHP can be embedded into a HTML document with a start tag and an end tag. It can be used on major operating systems such as Microsoft Windows, Linux and MAC OS X. It also supports web servers such as Apache, IIS and many more. It's both a object oriented and a procedural programming language and can be used in individual or mixed together.[20]

## 3.3.2   Usability Testing

The approach to evaluating user interfaces involves collecting data using a combination of methods, which could be experiments, observation, interviews and questionnaires - in a controlled setting. The primary goal is to determine whether an interface is usable by the intended user population to carry out the tasks for which it was designed, by investigating how typical users perform on typical tasks [12].

The purpose of usability testing is to evaluate a product or service by testing it with representative users [21]. A typical test is something like letting a participant try to complete some tasks while being observed. The observers will watch, listen and take notes. The goal of a test session is to identify any usability problems, as well as collecting qualitative and quantitative data. The data will then be used to determine the participant's satisfaction with the product.

By doing usability testing, problems can be identified concerning the design and development. This leads to issues found being identified and fixed before the coding and designing being made.

During a usability test, the observer will learn if participants are able to complete specified tasks successfully, and identify how long it takes to complete the specified tasks. By measuring the success of the participants, changes required to improve the user performance and satisfaction can be identified. To find out how satisfied the test users are with the product or service, a form with specific questions or a quick interview with a few questions about the test are often conducted. By analyzing the performance, together with the answers given for the questions, the owner of the test can see if it meets the usability objectives.

For a successful usability testing session, there should be a goal behind it. Usability testing is not just a milestone to be checked off on a project schedule. By conducting a test session, the holder of it should be ready to implement the results if necessary.

### Metrics

To determine the best interface using usability testing there has to be some metrics to use when comparing the two interfaces. A metric is a system or standard of measurement used to describe more than one attribute. It's important to have reliable measurements to use during evaluation of software, websites and other sorts of applications. [15, 18]

There are tons of metrics to use in a testing session. The main category of metrics are effectiveness, efficiency and satisfaction. Therefore at least one metric from each category would be a good way get a realistic picture about if the system is usable or not. [15, 18]

Each participant will be asked to save their configuration for later analysis.

**Effectiveness** - One way to measure effectiveness is to measure the success rate, sometimes called the completion rate. Using this metric determines whether users can perform the task or not. During this thesis' test session, the success rate will be measured by observation.

**Efficiency** - There are at least two common ways of measuring efficiency according to the time it takes for a participant to complete a task successfully. The methods used in the thesis are time based efficiency and relative efficiency. During this thesis' test

session, the efficiency will be measured and observed in form of the long time the participants will take to complete the test. The formulas used for calculating time based efficiency and overall relative efficiency are the following[15]:

$N$ = The total number of tasks (goals)

$R$ = The number of users

$n_{ij}$ = The result of task $i$ by user $j$; if the user successfully completes the task, then $n_{ij} = 1$, if not, then $n_{ij} = 0$

$t_{ij}$ = The time spent by user $j$ to complete task $i$. If the task is not successfully completed, then time is measured till the moment the user quits the task

$$\text{Time based efficiency} = \frac{\sum_{j=1}^{R} \sum_{i=1}^{N} \frac{n_{ij}}{t_{ij}}}{NR} \tag{3.1}$$

$$\text{Overall relative efficiency} = \frac{\sum_{j=1}^{R} \sum_{i=1}^{N} n_{ij} t_{ij}}{\sum_{j=1}^{R} \sum_{i=1}^{N} t_{ij}} \tag{3.2}$$

**Satisfaction** - measuring satisfaction can be done in at least two different types of levels, task level and test level. Both levels usually have some sort of questionnaire point system. The task level questionnaires should be answered right after each task and the test level questionnaires should be answered at the end of the test session. One example of a test level questionnaire is the System Usability Scale (SUS), described in detail in Chapter 3.3.3.

### 3.3.3   System Usability Scale

The System Usability Scale (SUS) provides a "quick and dirty" tool that is reliable which is used for measuring usability and satisfaction. [21] It was originally created by John Brooke in 1986 with the purpose of evaluation of a wide variety of products and services. The variation that the evaluation can be used for is i.e. hardware, software, mobile devices, websites and applications. SUS has become an industry standard for measuring usability with references in over 1300 articles and publications. It consists of a 10 item questionnaire with five response options for respondents. The five different options goes from strongly agree to strongly disagree. It is possible to alternate the questions from the original SUS to make the questions all positive but there were little evidence found that alternating items outweighed the disadvantages.[11]

There are some benefits of using a SUS as well as downsides. Some of the benefits are that it is a very easy scale to administer to participants and can be used on small sample sizes with reliable results. SUS is valid and it can effectively differentiate between usable and unusable systems.

Before deciding to go with a SUS evaluation, there are also some things to consider as it might not fit everyone. One has to keep in mind that the scoring system is somewhat

complex. The participant's scores for each question are converted to a new number, added together and then multiplied by 2.5 to convert the original scores of 0-40 to 0-100. There are two things to take into consideration when calculating the score of a participant. The first thing to keep in mind is if the question is on an even number or not. The second thing to take into consideration is the answer. The answer which can be given is from 1(Strongly agree) to 5(Strongly disagree). If the question is an uneven number, 1 point is deducted from the answer. If the question is an even number, the answer number is deducted from 5. Thereafter, the points for each question is added and the total score is multiplied by 2,5, which will land on a score between 0-100. Even though the scores are 0-100, these are not percentages and should be considered only in terms of their percentile ranking.

The best way to interpret the results involves "normalizing" the scores to produce a percentile ranking.[14] By doing so, a score can be measured in a proper way.

Based on research (point system), a SUS score above 68 would be considered above average and anything below 68 is considered below average. More details about the scoreboard in figure 3.2
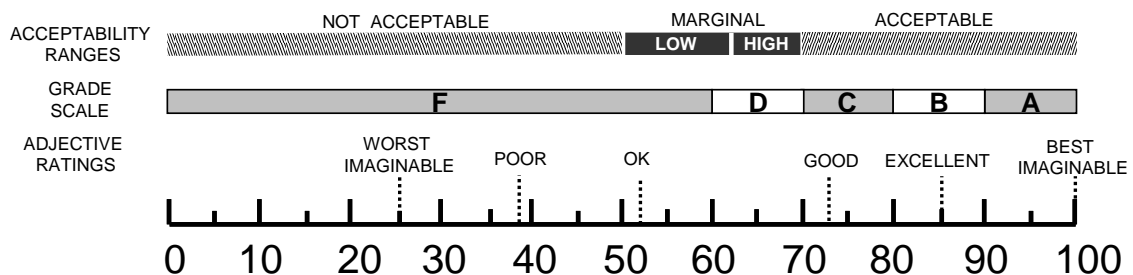


**Figure 3.2:** SUS score from Journal of Usability Studies [14]

When a SUS is used, participants are asked to score the following 10 statements with one of five responses that range from Strongly Agree to Strongly disagree[10]:

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.

# 3.4 Source Criticism

This section describes where the sources used in this thesis are taken from. Details about the sources can be found in the bibliography. The input given by the supervisors and other employees at Axis has been used in this thesis. They are being considered as trusted sources because of their experience in the company, its products and services. The purpose of their input was to validate and fulfill the goals as well as gathering information about the company for the thesis.

## 3.4.1 Literature

The literature used in the thesis are books recommended from the UX expert the thesis workers interviewed during the literature study. The first two books are being used in two different courses in LTH, Campus Helsingborg. The thesis workers considers these sources as trustful as it is being included in a higher form of education. The fourth source was recommended by an UX lead in Axis. The literature used are the following:

Yvonne Rogers Jenny Preece, Helen Sharp. *Interaction Design - Beyond human-computer interaction.* John-Wiley-and-Sons-Ltd, 2015.

Soren Lauesen. *Software Requirements: Styles and Techniques.* Addison-Wesley, 2002.

Kevin Matz. *Designing Usable Apps - An agile approach to User Experience Design.* Winchelsea-Press, 2013.

Jerry Cao. *The Guide to Mockups.* UX Pin, 2016

## 3.4.2 Internet

When open source tools are used for development, it is not always possible to only rely on books and literature as the framework is relatively new. Therefore, the thesis workers decided to use sources from the Internet where most sources regarding the technologies used are websites. These are used to complete the literature. The following sources are links found on the Internet and can be split up in two groups, official websites and unofficial websites.

### Official websites

The following websites are direct official websites for each technology used and are considered as trusted sources. The list includes Axis Communications, Balsamiq, Bootstrap, Datainspektionen, Usability.gov, jQuery and The PHP Group.

### Unofficial websites

There are also a few unofficial websites used for gathering information. These websites can be split into following categories, and complement each other in the area of user experience, which makes them somewhat trustworthy as they are used for the same purpose.

**UX Design**  - modernux.se, uxdesign.cc and usabilitynet.org

**Usability**  - uxpajournal.org, measuringu.com and usabilitygeek.com

**Web Development**  - w3schools.com and quirksmode.com

The websites under the category *UX Design* are websites relevant for gaining information about how a design should be made. The authors for the first two sites describes themselves as experienced in the UX design area as they have been working with the subject for a long time. The UsabilityNet website is mainly a project founded by European Union, and finished the development back in 2003 which means it can be considered as outdated. It was considered relevant by the thesis workers as it describes the ISO standards for certain measurements used for designs.

The websites under the category *Usability* are websites for gaining information about how to measure usability. The websites explains how the System Usability Scale works and why it is a standard in the world of UX. It was considered as trusted to the authors as the ones behind the Journal of Usability Studies are entitled principal members of technical staff and professor in the practice for Rice University in department of psychology.

The websites under the category *Web Development* are websites for gaining information about how web development works and how its background. The authors have considered w3schools.com to be a trusted source as it is an official goto website for everyone wanting to learn how to program some kind of programming language for the web. Quirksmode is a webpage developed by the freelancer Peter-Paul Koch with core expertise within browsers and web-development. Therefore, it is considered as a trusted source when it comes to JavaScript.

The article about the increase of camera surveillance in society is a news article from SVT which is considered as a trusted article since it is from a trusted channel in Sweden. The article is only used to validate the fact that surveillance in society is growing.

# Chapter 4

# Result Analysis

This chapter contains the results of the methods used during the thesis. It also contains the analysis of the design studio workshop as well as the interfaces and the usability tests on them. This chapter also contains a section where the presented results are discussed.

## 4.1  Mockups

There were two mockups made early in the thesis with the purpose of using them for the interviews. Getting opinions from the people being interviewed about the developed mockups was a vital part of the thesis work. The two mockups were made with different approaches. One of the mockups were made with the order of action, device, trigger (figure 4.1) while the other mockup is designed with the approach of device, trigger, action (figure 4.2).

The two different mockups were made to get a measurable result in the opinions about which approach seems to be the best and easiest from the participants point of view.

The two designs referred to as mockups are very similar to wireframes, but were left as mockups since wireframes generally only has a very plain design with only placeholders to show with no functionality what so ever. In the mockups used, there were some functionality mentioned and the purpose were to use them in the interviews as very early prototypes. This to get an idea of what the participants of the interviews would think of the produced designs. After analysis of the opinions of the mockups from the participants, the understanding and approach of this thesis grew. With the help of the opinions of the mockups, the current configuration settings became more understandable as well as what was desired in the future. [6]
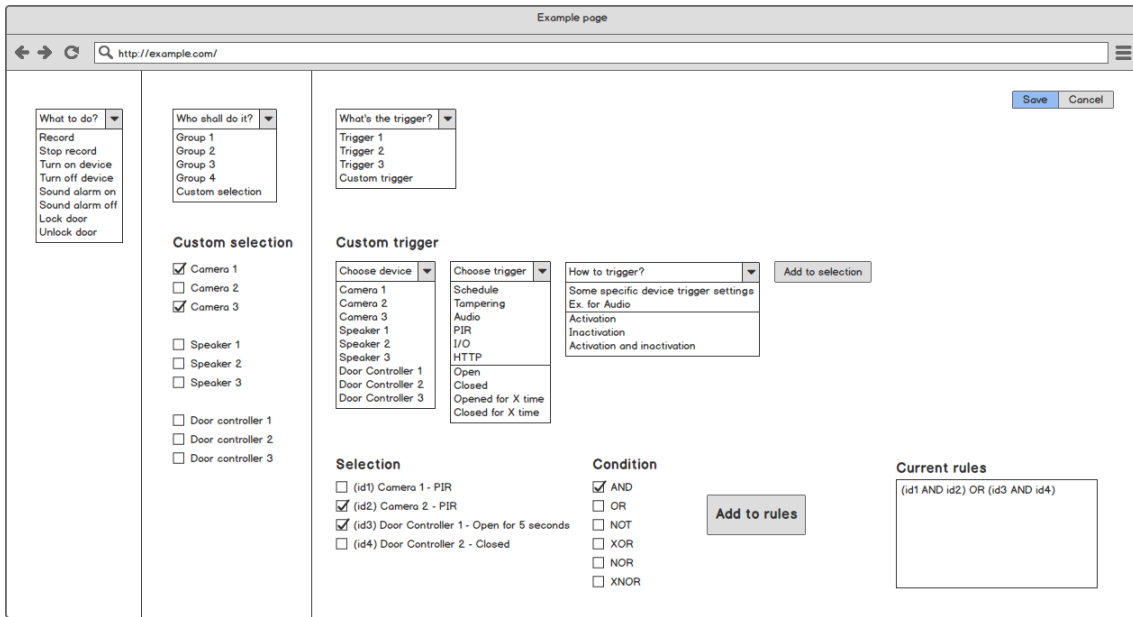
**Figure 4.1:** Mockup 1 used for interviews, with the configuration order action, device, trigger.
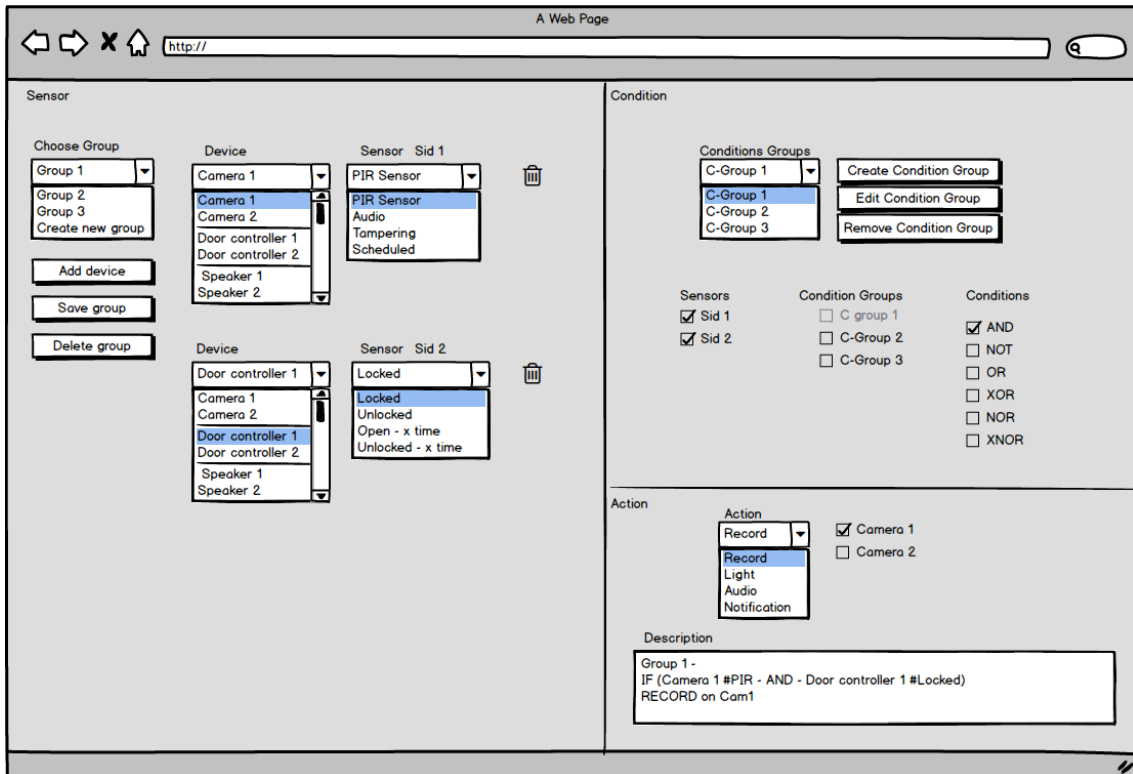


**Figure 4.2:** Mockup 2 used for interviews, with the configuration order device, trigger, action.

# 4.2 Interviews

To set the scope, all participants were asked how many devices they saw as realistic for this thesis to work with. All agreed to that AVHS is a system ideal for configuring two to six devices per site, with possibilities to configure up to 16 devices in it. When asking the participants about what was needed on the market, or what the clients wanted, all of the participants had a wish list of their own, but some of their requests were regarded as more important than others. The following features were chosen as they seemed relevant for the thesis, because they were being mentioned by many of the participants.

- One request, very popular between both participants and customers are relations between devices. This feature exists in the current interface but is very hard to set up and when configured, it's very fragile and easily broken(see Chapter 2.2.1 The specific relation building on "one-to-many" devices was discussed. This can be explained as the desire of being able to configure multiple actions for one trigger, which means that when something triggers, multiple actions are taken.

  A concrete example of this is a scenario like this: *A customer wants to identify a person from a camera when the keycard is used for a door. The system could compare the identity found by using the camera and the identity of the used keycard. If they match there is no need for alerts. But if they doesn't match, maybe the customer wants an alert of unusual access to the building.*

  This could prevent thefts accessing the building unseen. So it could be very useful for security reasons.

- Next thing on the wish list can be summed up in one word, simplicity. To reach a simple configuration, the use of wizards were thought of, which is a way to lead the user step by step to a specific configuration. Templates are also a way of helping the user to finish their configuration. A template is a set of predefined settings for a specific device or a scenario, saved into a file. This should be easy to configure so that it is possible for a user to do without any kind of education.

- Setting up devices in batch is wanted among the stakeholders at Axis. A suggestion to how this can be done is that the user makes a template and uses it to configure multiple devices with the same settings simultaneously. The purpose of this feature is to help many user groups to save time while configuring an event. Simplicity is something that is greatly preferred. If a configuration can be seen as simple and achieved by a few more clicks, that's great. To make it easier to configure the devices, templates might be relevant. The possibility for the user to configure the template to fulfill the needs of the user is important. Templates can be described as predefined settings for one or many specific devices. Another way to approach the problem is to configure one device, save the configuration and apply it on all the devices which needs the same configuration.

All persons asked think there is a desire from the customers to be able to configure multiple devices simultaneously, and it's an important and valuable feature, if it is presented to the customer in a simple way. Making it possible to configure multiple devices simultaneously, only for advanced users is not appealing to the customer according to the

interviewed persons. If it is possible to do it in a simple way for many devices, and leave the advanced settings to be handled on the few devices needed is a better solution. This is a problem today for many reasons. One reason being each device needs specific settings depending on what they are looking at, the device might not have an active trigger - but just waiting for a specific thing to happen or depending on what specifically it is looking for, the settings might be quite advanced. Another reason might be that all cameras have different triggers and maybe, for example one or two cameras purchased has a specific trigger. This means that only these two cameras has to be configured differently from the rest of the cameras, since only these two has the option to configure depending on a specific trigger.

It will always be more work to set up the configuration if there are more features. It's like a quota. If you have one device to work with and one feature available then we have a 1:1 ratio. If you add more devices and features you also add more work.

There is a common desire from the persons interviewed to see some kind of a clear overview of the devices containing relations to each other. There is also a common belief that all the needed information is not reaching the user in need of it, and hopes to get more information about what happened, and when it happened in a more clear way.

## 4.3 Brainstorming

There was a brainstorming session held to find scenarios reflecting situations that could occur in reality. The participants of the brainstorming session were the two thesis workers and a software developer. These situations are focused on an action to happen when something is triggered or the other way around, when something is triggered one or more actions are to be taken.

Many different scenarios were found with the main ones focusing on security breach, access to facilities, incident in a public area, guard tours and static monitoring. Every scenario were talked through which lead to deciding which triggers has high probability of being triggered, and which actions has a high probability of being needed.

By studying the scenarios, some use cases (see 4.4) could be formed. The use cases were edited in iteration to find the most suitable use case table.

## 4.4 Use Cases

The use cases were made by brainstorming situations or areas where security breaches could happen in. Different scenarios in the use cases were taken from the brainstorming session and developed further to become a complete use case. Every scenario was picked so that the area of use would not overlap with each other.

## 4.4.1 Break-in Protection

**Table 4.1:** Use case #1

| NAME | Motion in locked up facility |
|---|---|
| ID | UC1 |
| PURPOSE | The security manager wants to know when there is a movement in a locked up facility after closing time, and wants to be able to get an overview of what is happening in the facility when someone is inside. |
| SCENARIO | There is a store protected by three cameras. A person has somehow entered the store after closing time, camera #1 detects the movement of this person in the dark. Camera #1 starts to record, together with camera #2 and camera #3. A notification is sent to the security manager of the store. |
| CONFIG DESCRIPTION | The store is closed. One of the cameras detects movement with the help of its PIR sensor. All cameras in the facility starts to record and a notification is sent to the security in form of an e-mail. |
| CONFIG OVERVIEW (event config) | Triggers: Camera #1 (PIR) \| Camera #2 (PIR) \| Camera #3 (PIR) <br> Actions: Rec on camera #1 & Rec on camera #2 & Rec on camera #3 & Notification (E-mail) <br> Schedule: Specific |

## 4.4.2 Permitted Access to Area

**Table 4.2:** Use case #2

| NAME | Loitering outside facility |
|---|---|
| ID | UC2 |
| PURPOSE | The security manager wants to know when a person is loitering outside the facility to prevent future damage in the area at any time of the hour. |
| SCENARIO | There is a building with four cameras monitoring the surroundings of it. Someone is loitering outside and camera #1 detects the same person for five minutes, a notification is sent to security. When five minutes have passed, a sound clip is being played from camera #1 with vision of the person, and a lamp is lit on camera #1. |
| CONFIG DESCRIPTION | When someone is loitering outside at any time and any camera detects the same person for five minutes, a notification is sent to the security manager in form of an e-mail. Thereafter, a sound clip is being played as a warning from the speaker of the camera that has vision on the person loitering, as well as the LED lamp on the same camera is lit. |
| CONFIG OVERVIEW (event config) | Triggers: (Camera #1 (PiP) \| Camera #2 (PiP) \| Camera #3 (PiP) \| Camera #4 (PiP)) & Time (5 min)<br>Actions: Notification (E-mail) & TriggeredCamera (Speaker) & TriggeredCamera (LED)<br>Schedule: Always |

## 4.4.3   Expected Events

**Table 4.3:** Use case #3

| | |
|---|---|
| NAME | No movement outside the entrance within a specific timespan |
| ID | UC3 |
| PURPOSE | The security manager wants to know when an area that regularly is busy does not have any movement to be able to tell if something is wrong in it. |
| SCENARIO | A camera is installed outside the mall watching over the entrance area. The device detects no movement in the picture within a 30 minute period and a notification is sent to the security as well as a recording is started on the camera. |
| CONFIG DESCRIPTION | A camera detects no movement within 30 minutes and sends a notification in form of an e-mail to security and starts to record the area. |
| CONFIG OVERVIEW (event config) | Triggers: No motion detection & Time (30 min)<br>Actions: Notification (E-mail) & Rec on camera<br>Schedule: Specific |

## 4.4.4  Static Monitoring

**Table 4.4:** Use case #4

| | |
|---|---|
| NAME | Something changes in store window area |
| ID | UC4 |
| PURPOSE | The store manager wants to know when something changes in the store window, as well as a recording to see what might happen after the change. |
| SCENARIO | A camera is set up to watch over a store window area with the ability to compare pictures. The camera is set up to compare pictures every ten seconds. A mannequin has fallen to the ground and the camera detects the change. A notification is sent to the cash register and the store manager as well as a recording starts on the camera. |
| CONFIG DESCRIPTION | When difference in picture is noticed, notification is sent to the cash register with a HTTP notification and a notification is sent to the store manager in form of an e-mail. The camera in the store window starts to record. |
| CONFIG OVERVIEW (event config) | Triggers: Camera (DiP) & Time<br>Actions: Notification (HTTP-trigger) & Notification (E-mail) & Rec on camera<br>Schedule: Specific |

## 4.4.5  Incident in Public Area

**Table 4.5:** Use case #5

| NAME | Incident in public area |
|---|---|
| ID | UC5 |
| PURPOSE | The security operator wants to know when there is a loud voice detected or any rapid movement to assist when something happens is public areas. |
| SCENARIO | There is a public area monitored by two cameras. Someone gets a heart attack at the bus station and another person screams loudly. The loud scream is detected by the microphone on camera #1, a notification is sent to the security in form of an e-mail and a recording starts from camera #1 and camera #2 in the area. |
| CONFIG DESCRIPTION | A loud noise or a rapid movement is detected by the microphone on one of the cameras, a notification is sent to security in form of an e-mail. All cameras in the area starts to record. |

| CONFIG OVERVIEW (event config) | Triggers: | Camera #1 (Audio) \| Camera #2 (Audio) \| Camera #1 (VMD) \| Camera #2 (VMD) |
|---|---|---|
| | Actions: | Notification (E-mail) & Rec on camera #1 & Rec on camera #2 |
| | Schedule: | Always |

## 4.4.6   Monitoring of Emergency Exit

**Table 4.6:** Use case #6

| NAME | Emergency exit is used |
|---|---|
| ID | UC6 |
| PURPOSE | The security of a building wants to know, backtrack and set off an alarm when someone opens a door with the emergency exit handle instead of the regular handle during any time of the day. But it's okay to open the door without using the emergency exit handle. |
| SCENARIO | Someone is opening an emergency exit with the emergency handle at any time of the day. A notification is then sent to the security of the building, an alarm goes off and a camera starts to record video of the exit. |
| CONFIG DESCRIPTION | When the digital input from the emergency exit handle is pushed and the digital input from the door magnet is open, start an external siren through the digital output and start the recording of the camera at the door. Then send a notification to the security of the building in form of an e-mail. |

| CONFIG OVERVIEW (event config) | Triggers: | Digital input (Emergency exit handle) & Digital input (Door magnet) |
|---|---|---|
| | Actions: | Digital output (Siren) & Rec on camera & Notification (E-mail) |
| | Schedule: | Always |

# 4.5 Design Studio Workshop

Before it was possible to do the design studio workshop, the purpose of it had to be set. Deciding on what the workshop should be focused around and what the expected result would be were crucial for the thesis workers. The purpose of the workshop was set to decide if an order of how the interfaces would be designed were needed and if it were, what was the most natural order? With other words, how important is it to specify in which order the configuration has to be made? Does it have to be an order set which configuration needs to be done first between triggers, actions or devices? The second purpose was to create new ideas of how the interface could be designed. With the purpose of the workshop set, the result could be used as a foundation for future developments.

Six participants were chosen for the workshop, all of them working at Axis. Their roles were mixed between computer architect, software developer, project manager and UX developer. The participants were thereafter split into two groups with three participants in each group. Every participant were given a paper explaining the rules of the workshop. There were also a scenario a specific scenario explained on the paper for the participants to consider when designing their interfaces. Each iteration started with the participants getting a clean paper to draw on.

The workshop began with the thesis workers presenting themselves and the objective of their thesis, to further explain the purpose of the workshop. A short presentation was made about triggers, actions and devices, further to the importance of relations between them all. The participants were split up into two groups of three, and the rules of the workshop were set. The time limit for drawing a sketch was decided to five minutes, one minute for the presentation phase and two minutes for the critique phase. Permission were given by the participants for the thesis workers to audio record the workshop.

**Iteration 1** - The first iteration began with the participants being on hold for the first minutes of the iteration. It seemed like it was hard for the participants to translate the scenario into an interface, but came out with something to present at the end of the drawing phase. Everyone explained their sketch and a critique phase followed. Many of the comments in the critique phase were about comparing what was drawn to the interface of todays system.

**Iteration 2** - In the second iteration, the sketching started faster than in the previous one. The reason for that was as the design studio describes it, the group gets a better understanding of the problem as the workshop goes on. The deeper understanding comes from the participants talking about their ideas, and hearing critique of what could be done better and what could be added to the sketches.

**Iteration 3** - The third iteration sketches showcases that the idea of taking other peoples ideas and applying them on your own sketch is working. The development of the interfaces were moving forward and some features resembles what was presented in the previous iteration. The interfaces were starting to take form and were getting more complete and more complicated.

**Iteration 4** - The fourth iteration was the last one, some of the participants tried to make their sketches as complete as possible, while others tried to combine their old sketches with new ideas with the help of the comments from the previous critique phases.

**Post iteration phase** - After the fourth iteration, each group got a couple minutes to decide which sketch was the most complete one according to the scenario given at the beginning of the workshop. Any sketch of any iteration was allowed to be picked. After the groups picked a sketch each, they also got the possibility to improve the chosen sketch with their collective effort.

When a sketch was chosen by each group, they presented their new drawn sketch group wise and got critique from the other group on how they could improve it. Thereafter, the second group presented their sketch and a critique phase followed.

The final conclusion was that both sketches could have represented different parts of a system, or different phases. Both sketches could be combined to co-exist in a final system depending on what the user wants to accomplish.

# 4.6   User Interfaces

A literature study about user interfaces was done, with the goal of learning more about what kind of user interfaces that exist. In this period, an interview were made with a UX expert. The interview lead to the thesis workers thinking about conceptual designs. The idea of a conceptual design is that the person in question thinks of something specific, and then recreates that scenario in the interface (see Chapter 3.3).

One of the conceptual designs thought of was a ticket machine, where the user can go forward one step at a time with the goal of buying a ticket at the end. If something in the purchase needs to be changed, the whole process needs to be canceled and started from the beginning or the user can go back one step at a time. When translating this machine into interfaces, it is identical to a wizard.

Another conceptual design thought of was some kind of railway map where every option is visible from the beginning but only the user knows which way to take. The authors translated this into a graphical solution (see mockups in figure 4.3 and 4.4).

The interfaces are implemented for the web in HTML, CSS, JavaScript and PHP. The Bootstrap framework and the jQuery library are used as part of the interface functionality and design. A PHP script was written as a back-end handler to handle events such as saving and loading a configuration, templates, presets and other parts of functionality. The HTML pages were also generated with the help of PHP scripts. The event configurations are stored as JSON files on the PHP server.

The client side functionality, design, animations and pop-up windows are mainly written in JavaScript with the use of the Bootstrap framework, the jQuery library and the jQuery UI library.

Both of the interfaces are developed for Google Chrome web browser using a screen resolution of 1920x1080. They aren't scalable in a correct way in higher or lower resolution.

The source code, screenshots and mockups for the developed interfaces including the usability test can be located at GitHub: https://github.com/MehmetSimon/Examensarbete/
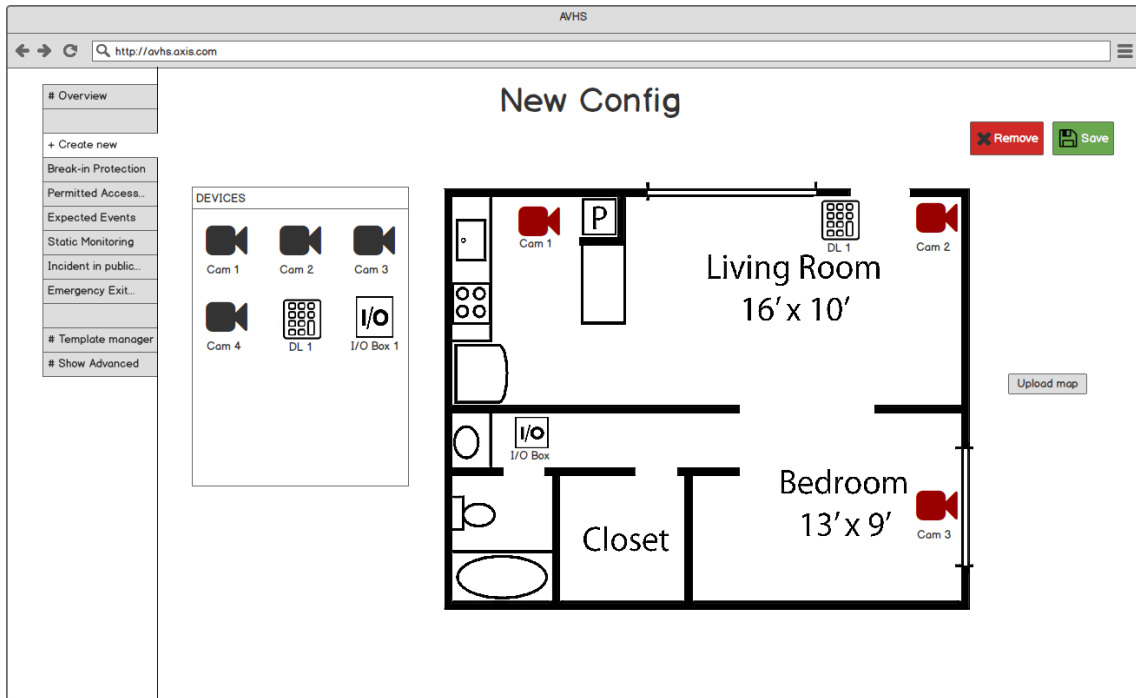
**Figure 4.3:** Mockup for interface 1 inspired by the design studio workshop.
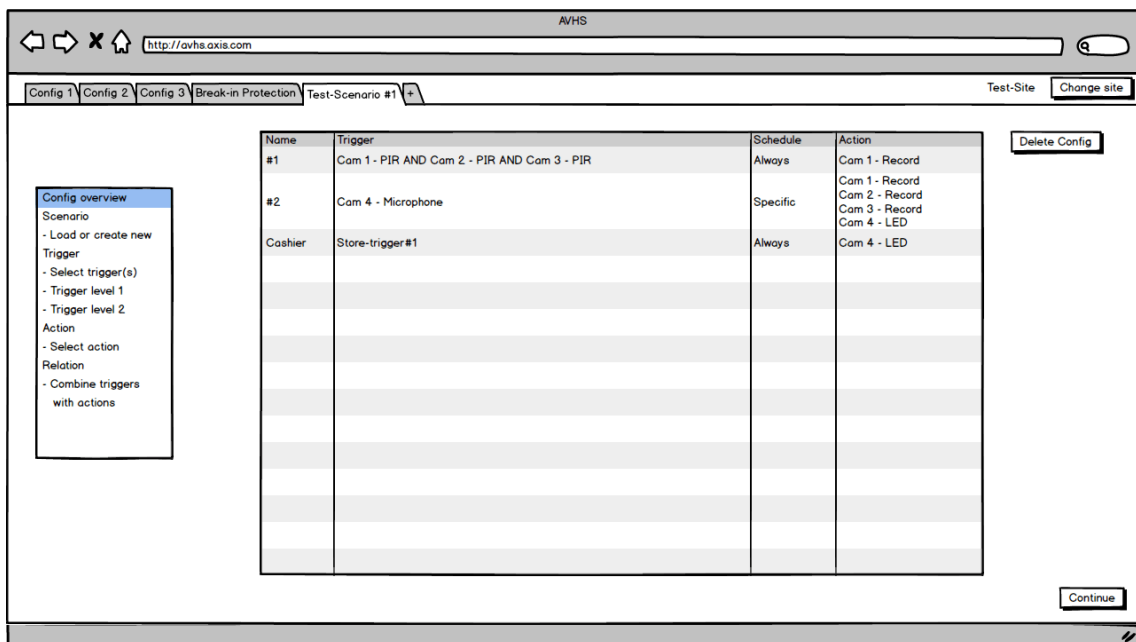


**Figure 4.4:** Mockup for interface 2 inspired by the design studio workshop.

## 4.6.1   Interface 1

Interface 1 is a combination of the authors ideas of conceptual designs, specifically the railway map, and the result of the design studio workshop. The configuration flow can be described as in figure 4.5
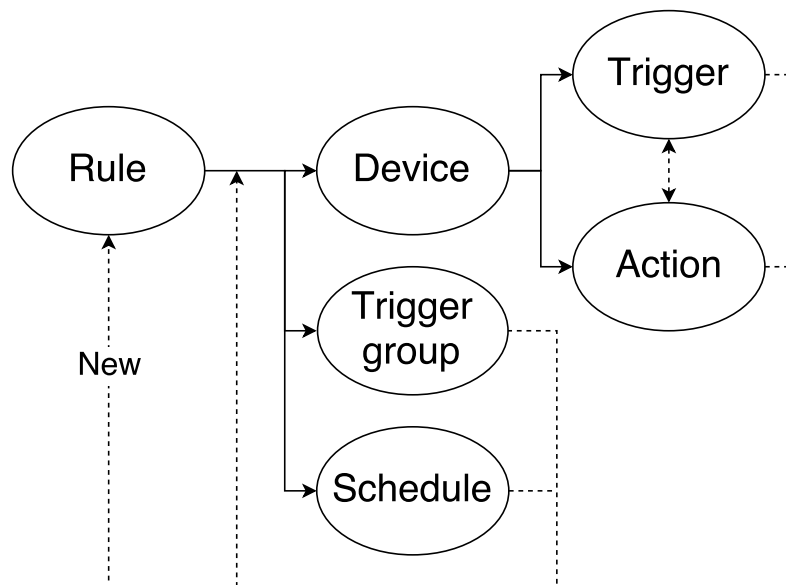


**Figure 4.5:** The flowchart for interface 1

A free to use template was used and edited to fit the needs of a sign up page. The sign up page was only used for the usability test and isn't of importance for this thesis. A screenshot of the page is shown in figure 4.6 below.
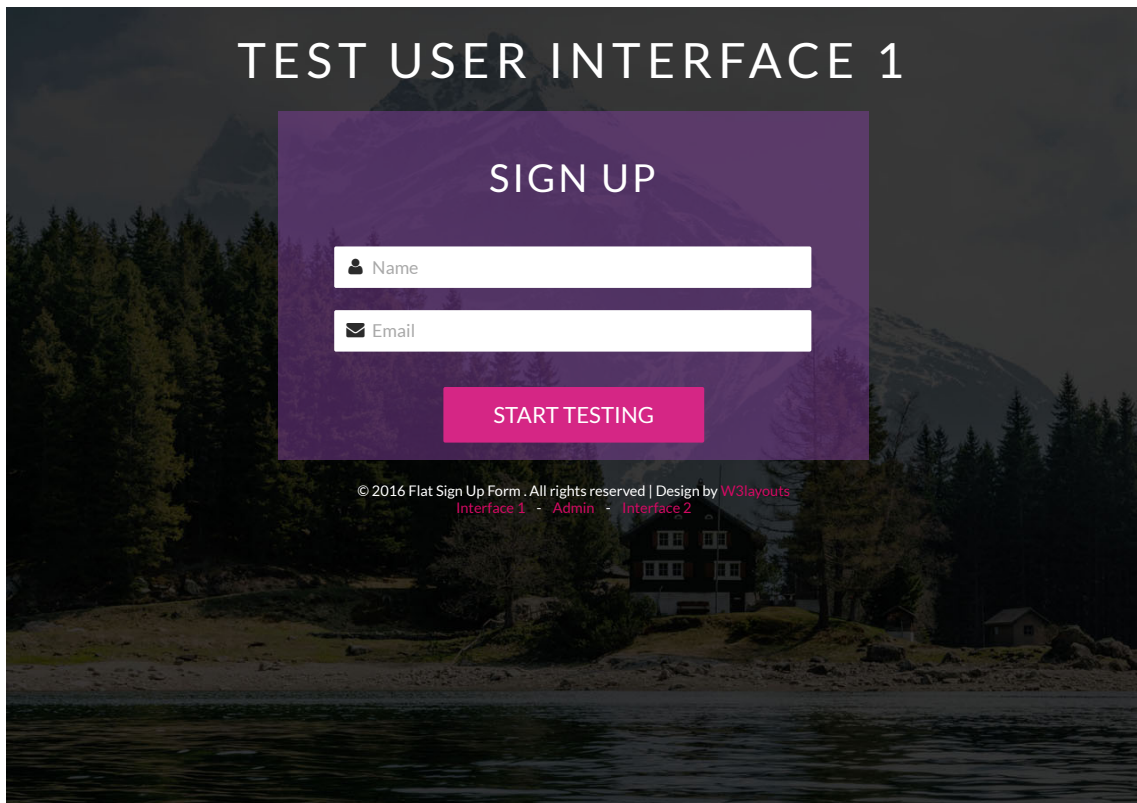
**Figure 4.6:** Test session sign up page for interface 1

There are two views of interface 1. It's the simple view and the advanced view. The simple view's main concept is to get a quick and relatively simple overview about the event configuration rules and to add pre-configured presets in a fast and simple way but also to remove rules. The rules are summarized using a rule name and icons of the configured triggers and actions for each rule.

It's also possible to save, delete and rename an event configuration in the simple view. An empty configuration in the simple view mode is shown in figure 4.7 and in figure 4.8 the use case in Chapter 4.4.1 is configured. It's possible to access saved event configurations, create a new configuration, access a template manager and switch between advanced view and simple view in the menu to the left side of the page.
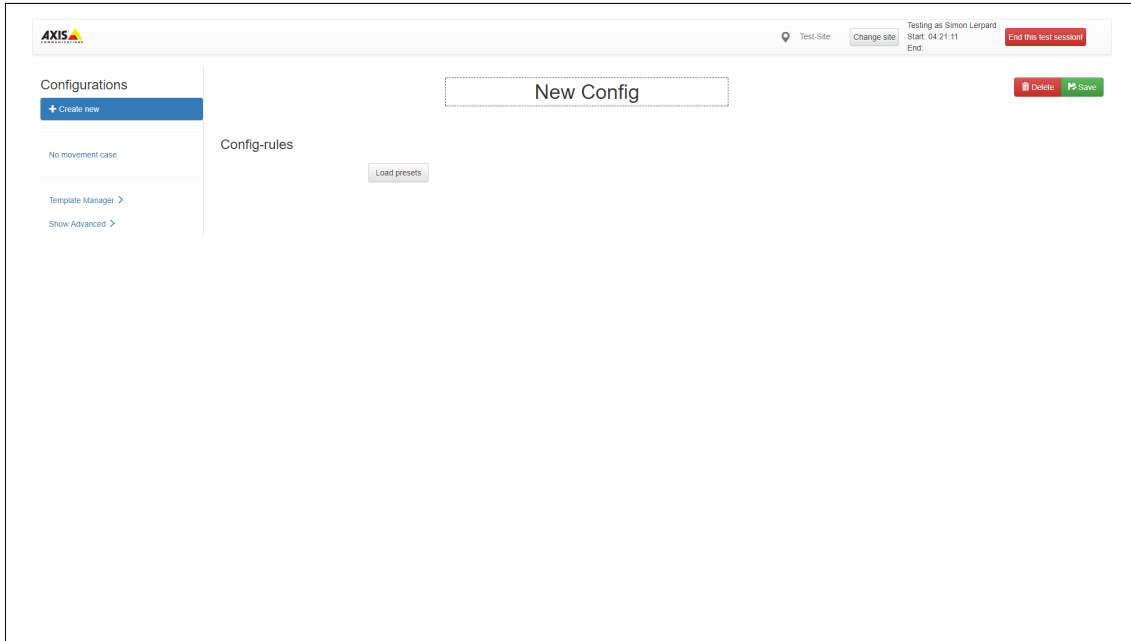
**Figure 4.7:** Interface 1 Simple view, Start page
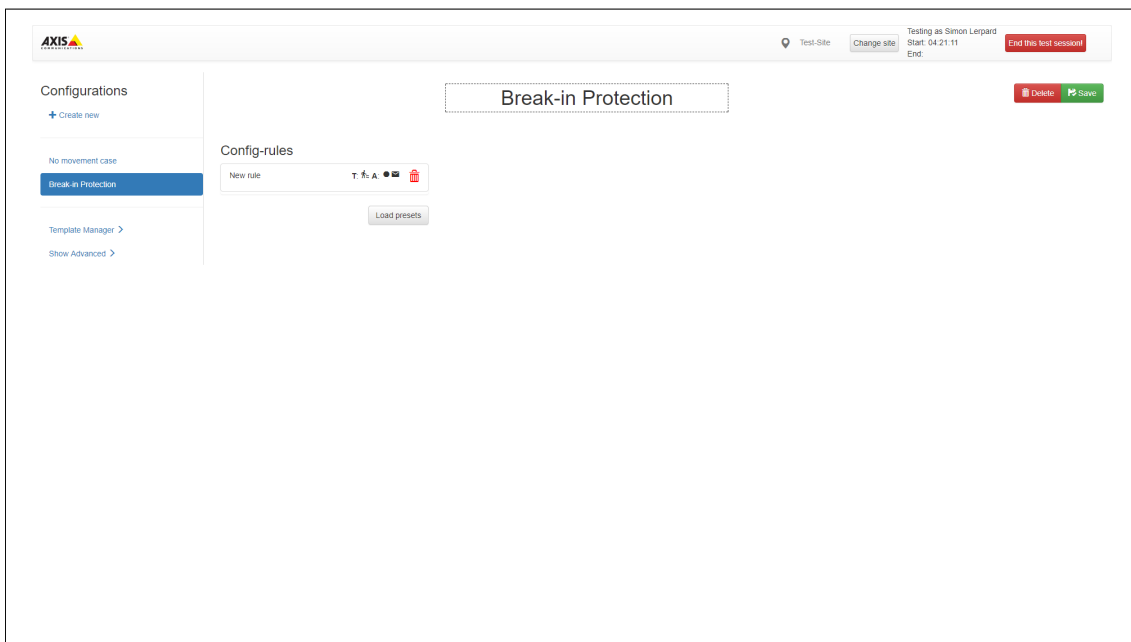


**Figure 4.8:** Interface 1 Use Case 1, Simple view

For the advanced view shown in figure 4.9, there are a lot more functionality compared to the simple view. It's possible to add a trigger and action from the listed devices/virtual devices. If a device icon is clicked, a sub-menu appears containing of the available actions and triggers for the device. If an item from the sub-menu is clicked on, it'll be added to the green marked rule. It's also possible to add groups to the triggers section inside the rule by clicking on one of the virtual devices All or Any. To add a trigger to a group

it's just as simple as dragging it inside the grouping box and drop it. It's also possible to double-click on a trigger to replace its device with another device that has the same trigger. For example, if a cam1 trigger is double-clicked a pop-up window is shown to replace the camera with another camera.

To create or update a preset, the button named *Save* to the right side of the rule below the *Preset control* headline needs to be clicked. A pop-up window appears to save a new preset or replace another preset as the selected rule. To load a preset to a rule it's just as simple as clicking on the load button to the right side of the rule. A pop-up window containing a list of presets will appear exactly as if the *Load preset* button is pressed in the simple view.

It's also possible to copy and paste a rule's actions and triggers by pressing the corresponding buttons to the right side of the desired rule. If the green plus (+) is pressed a new rule is created or if the red minus (-) button is pressed the desired rule is removed. Triggers can also be dragged and dropped between rules.

By pressing the *Edit schedule* button, a pop-up window appears and the option to specify the schedule is shown. Because of a few limitations of this prototype it's only possible to choose between *Always on* and *Specific*. The pop-up is supposed to show a more advanced schedule manager in possible future improvements.

The template manager, change site and the upload floor plan functionality isn't implemented for this prototype.
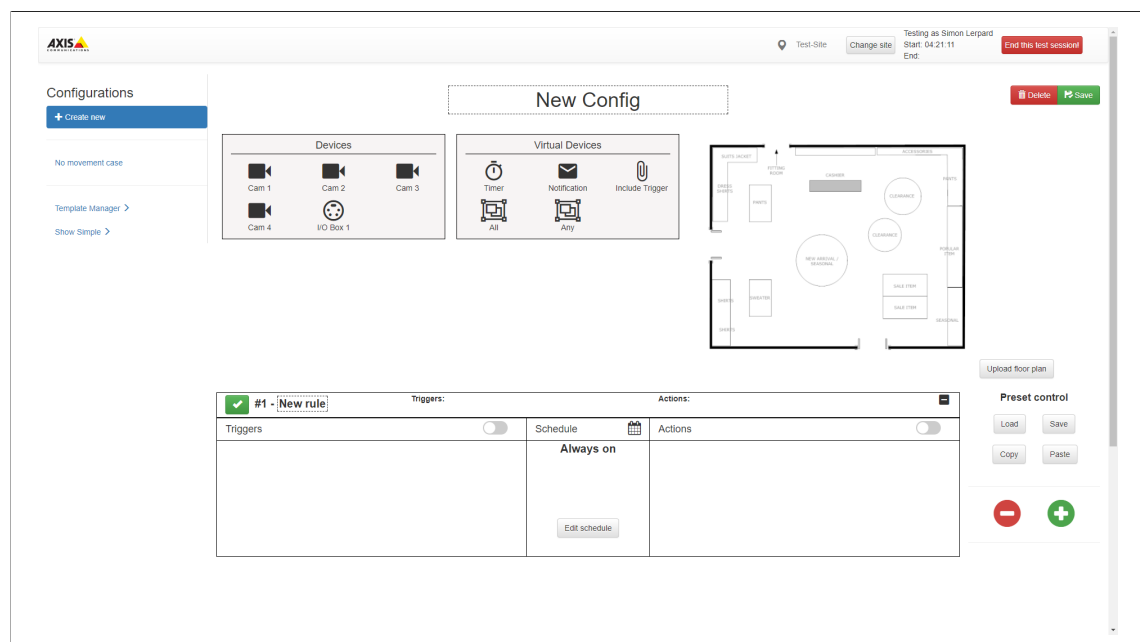


**Figure 4.9:** Interface 1 Advanced view, Start page

An enlarged screenshot of the advanced view for use case 4.4.1 can be found in Appendix C.1

## 4.6.2  Interface 2

Interface 2 is also a combination of the conceptual designs, but more specifically the ticket machine, and the result of the design studio workshop. The configuration flow can be described as in figure 4.10
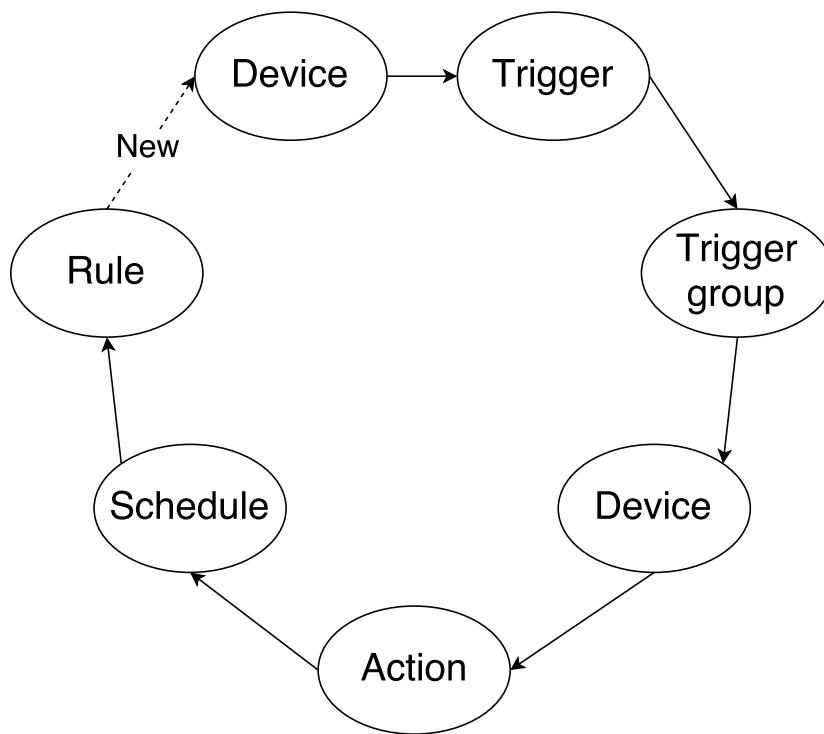
**Figure 4.10:** The flowchart for interface 2

There is also a sign up page for interface 2, shown in figure 4.11. The only visual differences from interface 1 is the title.
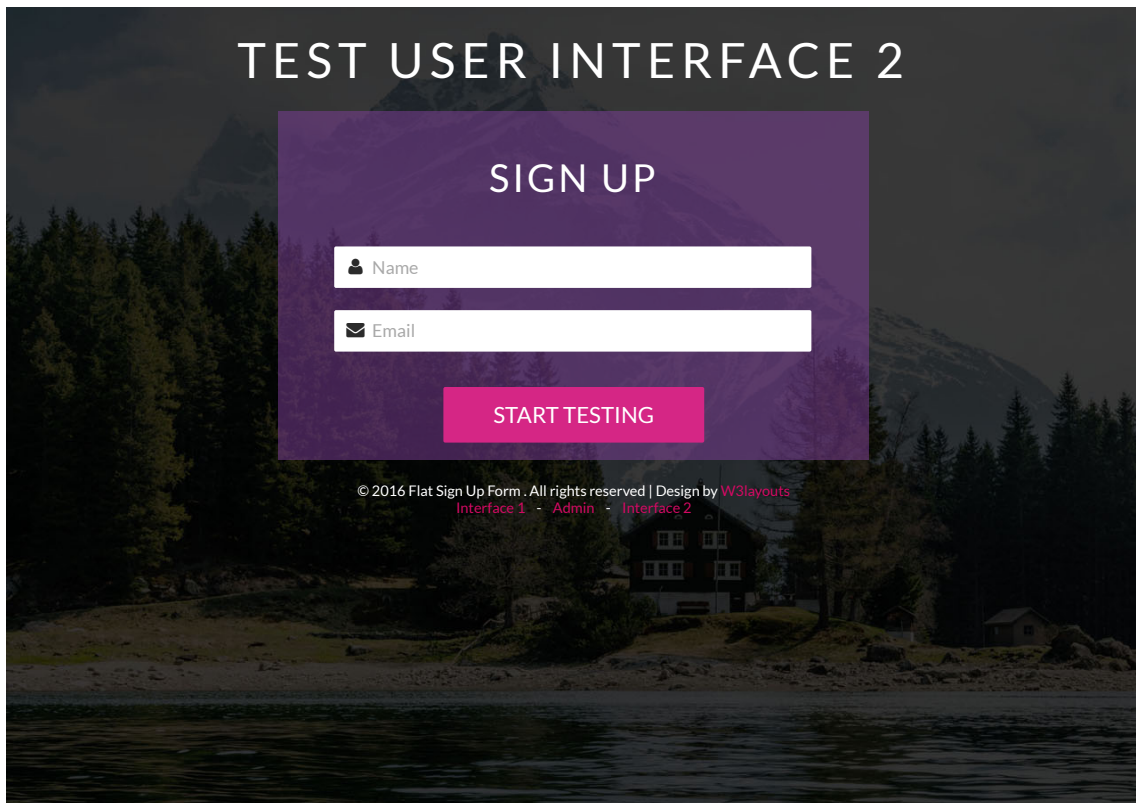
**Figure 4.11:** Test session sign up page for interface 2

The main concept of Interface 2 is the step-by-step method. The design is developed as a wizard where the configuration is done step by step. There is a navigation menu to the left side of the page to navigate and to get a good overview of the configuration process.

**Overview** - The first step shows an overview of the configured event configuration. It's possible to change the name of the event configuration from the overview page. It's also possible to save or delete the configuration from the server. Each event configuration is located at the top navigation menu, labeled by its name. A clean configuration page is shown in figure 4.12. Figure 4.13 shows how a configuration made according to use case 4.4.1 would look like.

**Scenario** - The configuration process is straight forward. There is a possibility to load a pre-saved scenario and the user can choose whether to load a chosen scenario or create a new one. A pre-saved scenario can also be referred to as a template.

**Trigger** - Triggers can be added in the next step by selecting a device. When the device is selected, a number of triggers can be selected in an interactive checkbox menu. Multiple triggers can be added at the same time by checking multiple checkboxes.

If the user wants to, it is possible to group triggers chosen. All the triggers added will show up as a list of checkbox items in the following step. By selecting the desired checkboxes and a grouping option, the triggers will be grouped as selected.

**Action** - The following tab is for adding actions for the configuration. When the device is selected, a number of actions can be selected in an interactive checkbox menu. Multiple actions can be added at the same time by checking multiple checkboxes.

**Relation** - Configuration of the relation between the triggers and the actions are made in the last step. This step can also be referred to as a rule creation step. A dropdown menu is shown in this tab containing of preselected triggers/groups from the trigger tab. Actions selected from previous steps are also shown as a list of checkboxes. To create a rule, a trigger has to be selected in the dropdown and the desired actions should be checked. A rule can be tagged with a name or left blank. When the button *And* is pressed, a pop-up window appears as the last step. The type of schedule that should be applied to the rule is chosen. When the schedule is set, the rule is created.
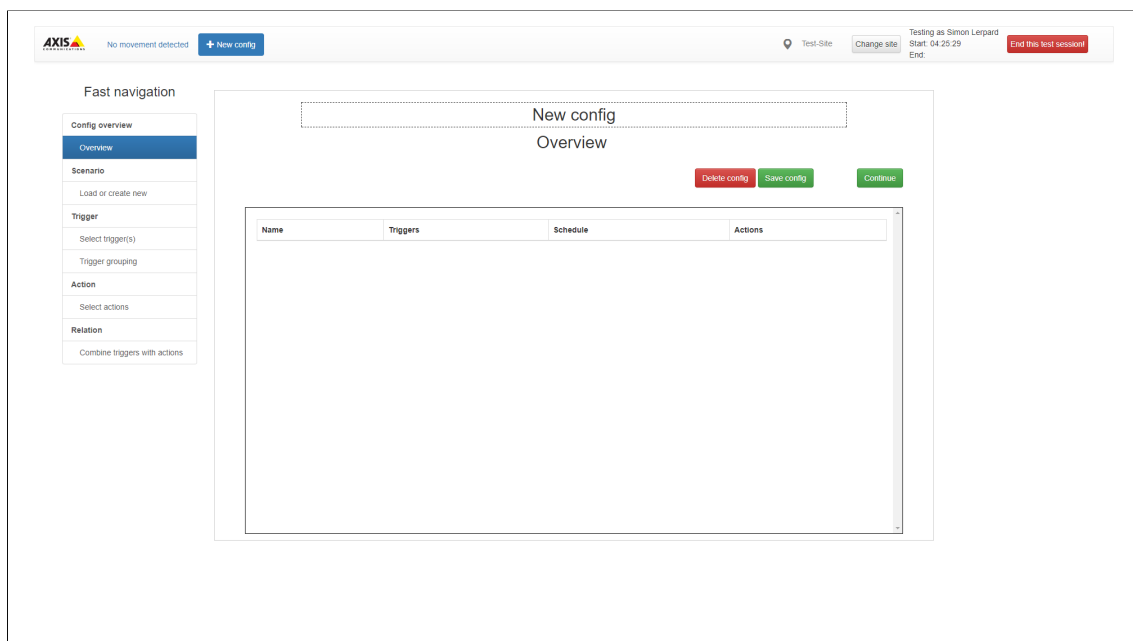


**Figure 4.12:** Interface 2 start page
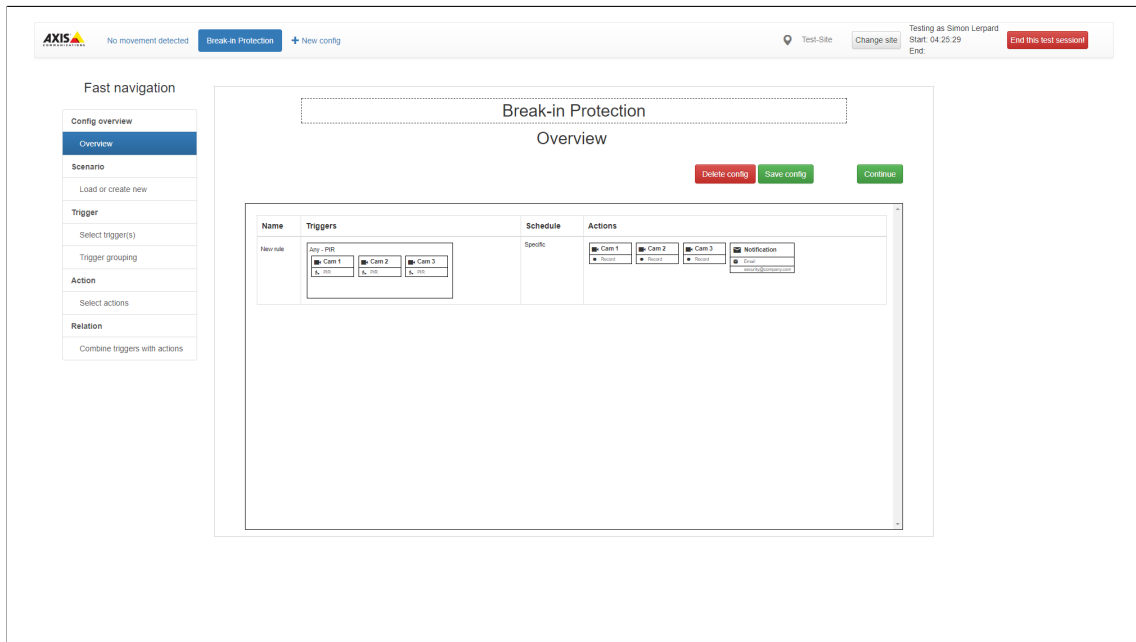
**Figure 4.13:** Interface 2 Use Case 1

An enlarged version of figure 4.13 can be found in Appendix C.2.

### 4.6.3 Usability Testing

Usability testing was conducted on a total of 14 workers on Axis, and their roles differs greatly. The testers of the interfaces consisted of software developers, student testers, summer workers, engineers and experienced engineers. The purpose of picking testers with different roles was to let workers which were not so familiar with AVHS test the interfaces, as well as workers very familiar with AVHS. The testing sessions were held in a room with two computers. When the testers arrived to the session in pairs, they got to sit next to a computer each, were handed three different tasks chosen from the use cases and a time limit to complete the tasks. The participants were given the opportunity to ask questions freely during the test session. The testers started with task one 4.4.1, continued with task two 4.4.5 and finished with task three 4.4.6. If there were time left, a fourth task 4.4.3 were given for the users to configure, as it was a bit more complicated than the previous use cases.

The purpose of giving the fourth task later than the initial three ones was that the testers had gotten some configuration experience with the help of the previous three tasks. The thesis workers were observing the participants while doing their configurations, as well as taking notes and answering questions.

When the test session was over, a SUS form prepared by the thesis workers were requested to be filled from the participants.

## Interface 1

The users felt the interface 1 to be somewhat inconsistent in design and layout as well as the words and terms chosen for the different features. The lack of help in form of tooltips or some kind of short description were one of the most given feedback on how the interface could be improved. The test users felt that they were confused by being presented the simple view page at the start of the test sessions of user interface 1. They did not know where to go from the simple view page. It would be appreciated if there were some kind of help in form of a hint, which could be one of the previously mentioned methods to the configuration page. This lead to the test users having a hard time finding the configuration page which was the advanced view. There were also some confusion with the coloring in the interface as a green button was mistaken as a save button when it was not. It was a bit unclear how to pick a trigger or an action for a device, and a bit of help on how to configure these would be appreciated. The drag and drop feature were not used a lot, as well as the grouping of the devices. If the test user did not pick an *all* box, the relation between the devices picked in the trigger box were set to be *any* automatically, which was neglected by the test users. There were some confusion with the interaction of clicking on an item that would appear somewhere else in the screen. Many of the test users had issues with going from simple view to advanced view, which created a lot of confusion and insecurity. The location for the *show advanced* button felt somehow hidden. Some kind of help on how to switch from simple view to advanced view would have made their test session easier.

## Interface 2

There were too much text according to the users testing the interface. They would have wished for a more "self explanatory" design as it was expressed. Explanation of functionality was missing according to the testers in form of tooltips on icons or some kind of description for them. Many felt that an undo button would have helped. When a configuration were made, the option to delete that configuration did not exist since the time to implement that feature were not enough before the test sessions started. This caused some confusion between the tests but were explained easily. The use cases presented were also hard to interpret for some of the test users, as they did not understand what should be configured sometimes. The flow from beginning to end of a configuration did not feel natural for many test users.

## 4.6.4   Measurements

### Effectiveness

The effectiveness of the test subjects were measured as the success rate of correctly configured use cases in respective interface, which can be seen in figure 4.14. The success rate for the first task in interface 1 was 85,7%, while interface 2 had a success rate of 42,9%.

The success rate for the second task in interface 1 was 71,4%, while interface 2 had a success rate of 57,1%.

The success rate for the third task in interface 1 was 71,4%, while interface 2 had a success rate of 42,9%.

The success rate for the fourth task in interface 1 was 28,6%, while interface 2 had a success rate of 14,3%.
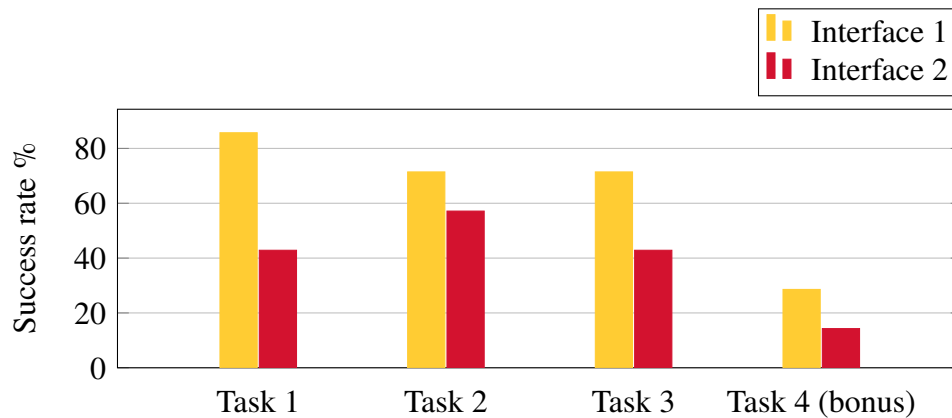


**Figure 4.14:** Success rate for how many percentage of the test users which successfully finished a task per interface.
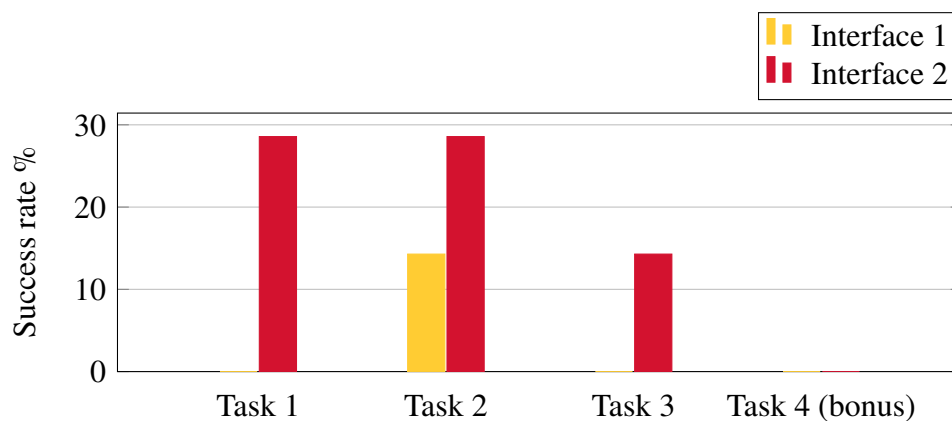


**Figure 4.15:** Success rate for how many percentage of the test users which were close to making a correct configuration. The relation was correct but some device were missing.
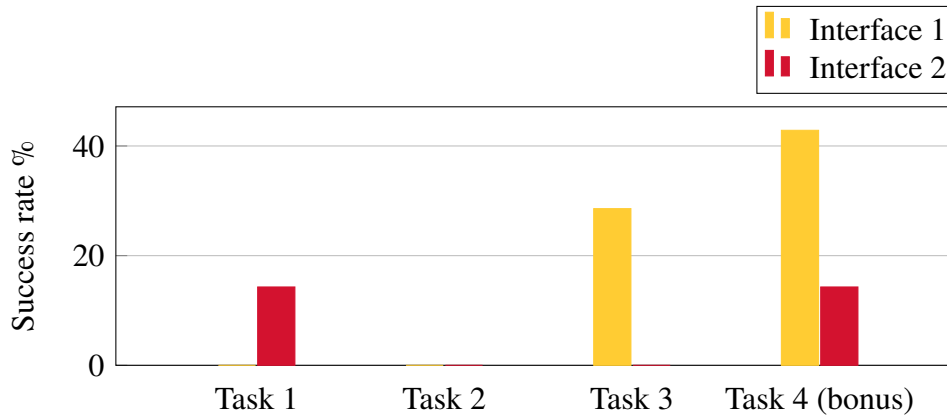
**Figure 4.16:** Success rate for how many percentage of the test users which had the correct devices configured but the relation between them wasn't correct.
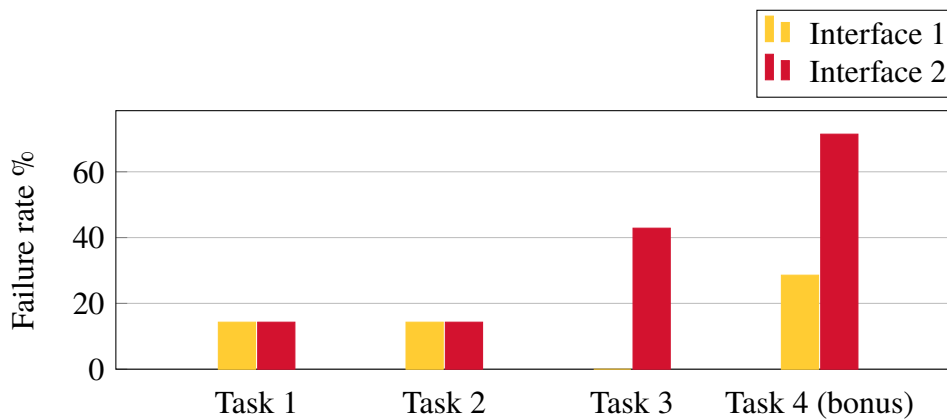


**Figure 4.17:** Failure rate for how many percentage of the test users which configuration was very incorrect or didn't exist at all.

## Efficiency

**Time based efficiency.** After analysis of the interfaces, there is not a big difference in how many tests finished per hour according to figure 4.18. In numbers, interface 1 will finish 4,3 tests per hour while interface 2 will finish 3 tests per hour. This gives us a difference of 43% in time efficiency in advantage of interface 1. The same table also measures how many of these tests made are configured successfully. When looking at the success rate, the differences are more distinct. The numbers shows that for interface 1 - 3 of the 4,3 tests made were configured successfully, while for interface 2 - 0,5 of the 3 tests made were configured successfully. This shows **that interface 1 were configured successfully with a 69,7% accuracy while interface 2 were configured successfully with a 19% accuracy.** Interface 1 is also 500% more time efficient when considering the successfully finished tests compared to interface 2.
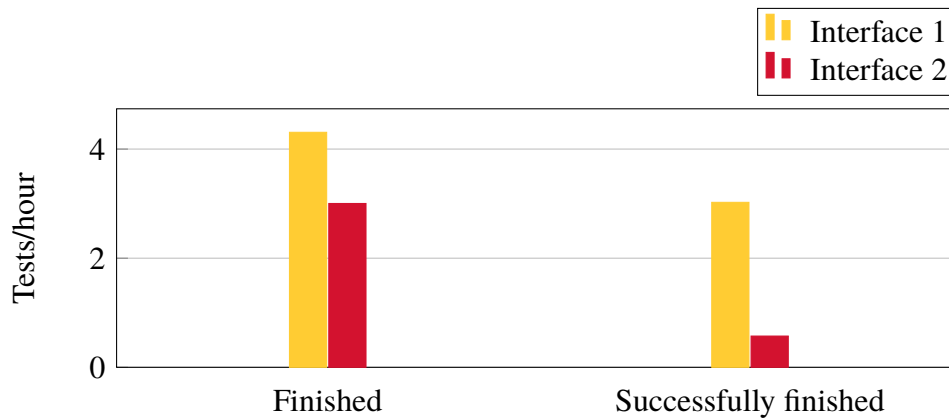
**Figure 4.18:** Time based efficiency. How many tests finished and how many tests configured successfully for each interface per hour.

**Overall relative efficiency.** After analysis of the interfaces, the thesis workers could see the finished test results being closer to each other. The overall relative efficiency for the finished tests for interface 1 were 79,24% and 83,88% for interface 2. These numbers does not show the whole truth tho. After analysis of successfully finished tests, the overall relative efficiency shows that interface 1 gets a score of 46% while interface 2 gets a score of 12%. When comparing the relative efficiency between interface 1 and interface 2 for the successfully finished tests, interface 1 is 283% more relative time efficient.
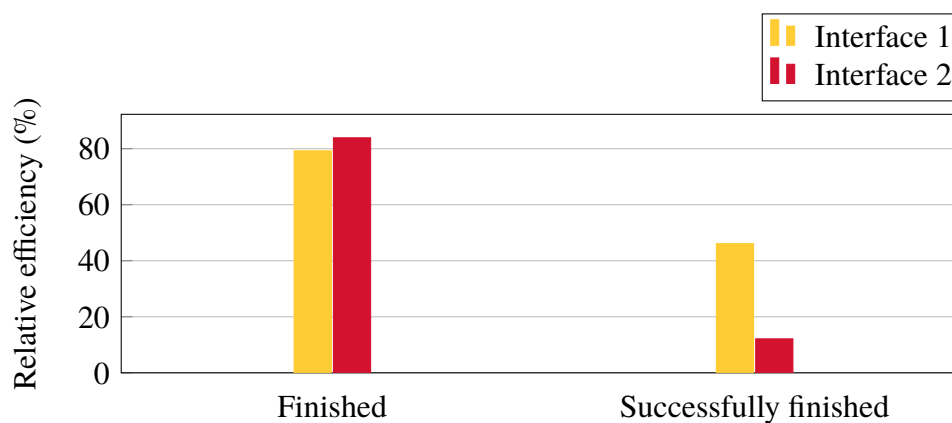


**Figure 4.19:** Overall relative efficiency. The ratio of the time taken by the users who successfully completed the task in relation to the total time taken by all users.

## Satisfaction

The satisfaction of the two interfaces were next in line to be measured by the thesis workers. Figure 4.21 shows the average satisfaction points based on groups of students, employees and a total of the participants. Students scored interface 1 to 40 points, compared to interface 2 which got a score of 55. When looking at the employees, the difference were

not quite as big, as interface 1 got a score of 58 compared to the 61 of interface 2. When measuring satisfaction for the total of all participants, interface 2 still gets more points, interface 1 gets a total score of 53, while interface 2 gets a total score of 58.
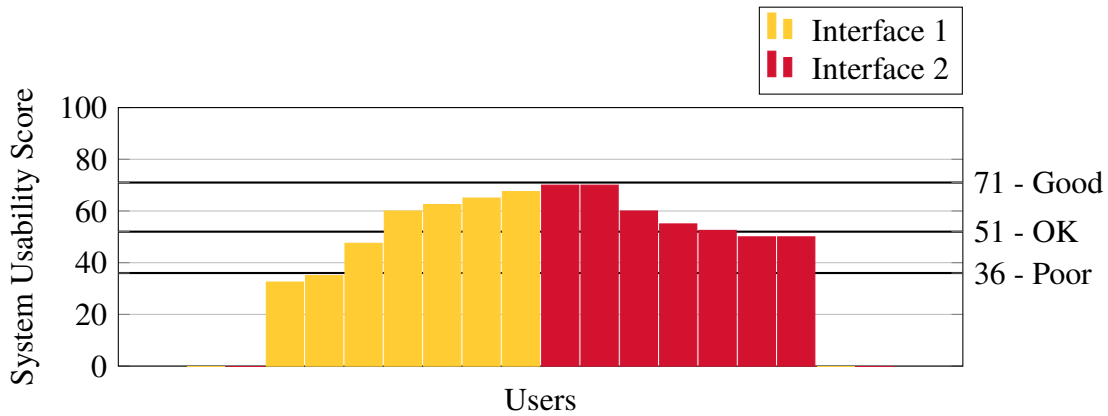


**Figure 4.20:** The System Usability Score for all test users for each interface.
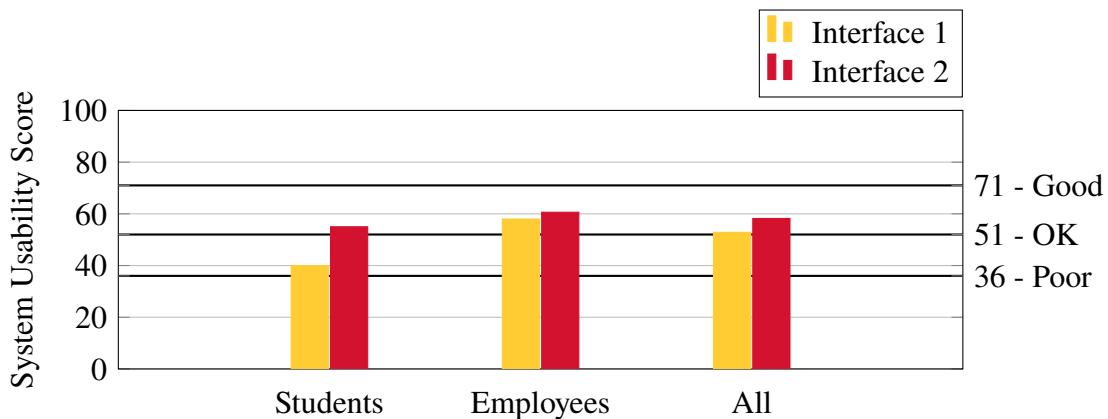


**Figure 4.21:** The average System Usability Score for each interface divided in three groups. Students (includes summer workers too), Employee and All.

# 4.7 Discussion

During the thesis, some parts of it stood out to the authors where they could see results worth to discuss.

## 4.7.1 Interviews

During the interviews, the test mockups shown to the participants, and their opinions of them were interesting as they showed how they want an interface to be designed. In the end, both of the mockups were seen as too complicated, and there was a desire for an

interface with less complicated configurations and simpler layout. The reason behind this request was that most of the devices would have their triggers configured in the same way, the ratio between simple settings and advanced settings were about 80/20. In other words, if the most common settings could be pinned down, about 80% of the settings desired would be solved. Another important thing to note was that most of the participants wanted to think in such a way that it always started with a scenario. An example of a scenario could be that a window or a door needs to be broken, which means that something needs to happen before triggering an action on a device. Therefore, a brainstorming session were set up to find relevant scenarios for the purpose of configuration.

## 4.7.2   Design Studio Workshop

The thesis workers put a lot of thought on the workshop's purpose and with which method it should use. In the beginning, card sorting was considered as a method for the workshop, to find out in which order the configuration should be done in the interfaces. Should the configuration of device, trigger or action be first or should the configuration be based on a scenario first and then set up the triggers, actions and devices?

The order of configuration can be traced back to the interviews where the participants had a mixed opinion about which configuration should be made first. The order of configuration was unclear, and the thesis workers asked themselves if the need to specify a set order was necessary. The purpose of the workshop were changed to investigate if an order were necessary, by sketching a configuration for a prepared scenario. When sketching an interface to fulfill the configuration of the scenario, the participants were given the chance to present how they wanted the configuration interface to look like and which features it should contain.

Another thing discussed among the thesis workers were the length of the workshop. According to the method, it is up to the responsible ones for the workshop to set the length. The number of iterations held are crucial, with the reason of having too few iterations can lead to the workshop not producing any constructive result, while having too many iterations can result in the workshop going too much in to detail about specific solutions. The desired result is somewhere in the middle, the participants should produce productive results and many different ideas which can be improved later on. To fulfill this condition, there is no need for many iterations. Therefore the authors decided that four iterations were enough for the workshop as each iteration were quite consuming. After the fourth iteration, the group was given time to discuss and agree on one sketch they found most promising, and run one extra iteration on that specific sketch to improve it together as a group.

## 4.7.3   Creating the Interfaces

The interfaces took a while longer to implement than planned and some functionality such as the template manager in interface 1 needed to be scoped out from the original plan. And because interface 1 took longer to implement, time was more critical and interface 2 didn't really get that good as planned. And because of that there was some lack of important functionality, such as edit or remove an added rule. This and other temporary solutions did definitely have some effect on the usability test.

For interface 1 the original plan was to drag and drop a trigger or action, but also because of time optimization, implementation simplicity needed to be prioritized so it got changed to a more clickable menu instead. The structure of the source code didn't get as good as desired too either.

## 4.7.4   Usability Test Results

After the testing session for the two interfaces, the results were measured with the help of the metrics mentioned earlier together with the result from the SUS. The test subjects consisted of nine Axis employees and five student workers, a total of 14 test subjects. The authors were expecting a higher number of participation for the usability tests but the numbers were held low due to lack of time and employees being on vacation. With a high number of the testers being student workers, the test results showed a difference between the employees which were accustomed to AVHS, compared to the summer workers which were not as used to AVHS.

There were three kinds of aspects taken into consideration then analyzing the usability test results: effectiveness, efficiency and satisfaction.

### Effectiveness

Considering the following numbers from figure 4.14 for each interface, the percentage from the success rate table varies a lot. Interface 1 is 50% more effective in task 1, 21% more effective in task 2, 40% more effective in task 3 and 50% more effective in task 4 than interface 2. This means that interface 1 is in average, 40% more effective than interface 2 in total of all tasks tested in the usability tests.

### Efficiency

When the time efficiency for the two interfaces were measured and translated into two tables which can be seen in Chapter 4.6.4 . There are two things to take into consideration when drawing a conclusion of which interface were the most efficient in aspect of time, which is time based efficiency and overall relative efficiency.

### Satisfaction

Numbers from figure 4.21 shows that interface 2 was more popular among the student workers by 17%. Among the Axis employees, the difference was only 4% in favor of interface 2. When measuring the total satisfaction among all the testers, the participants were more satisfied with interface 2 than interface 1 by 9%.

Based on the score table from chapter 3 in figure 3.2, the conclusion of both interfaces being *OK* can be drawn as the point needed for that is 51 or more, but none of them reaches a score needed to become an interface rated above average, which needs a score of 68 or higher.

## Observations

By observing the test users during each test session, the thesis workers gathered a lot of information. There were a lot of mistakes for both interfaces made by the test users which generated some good feedback. The observation data showed that interface 1 seemed to have a faster learning curve than interface 2, which could be related to its design. To have every option needed to create a complete configuration at the same page as an advanced overview could be a positive solution. The interface had some blank pages which could have made the user confused, and made the second interface harder to understand. Analysis showed that students were more bothered by this than the employees.

The behavior of the test users observed were very similar to each other in terms of interaction and exploration of the interfaces. The most common misunderstanding in interface 1 was the location of the advanced view tab. A big part of the test users had clicked on every button available in the simple view before the advanced view were found. The advanced view was critical to find in the process of completing the test as all configuration tasks was supposed to be made in it.

There were also observations of interface 1 over the drag and drop functionality. Only a few test users actually used the drag and drop function as it was meant to be used. Some test users tried to drag and drop items in the interfaces which were not dragable, which also lead to some confusion. A great deal of the test users tried to drag a device icon from the device list box in interface 1, but did not try to drag the created trigger until an empty group was added.

The last bonus task in the test session given to the test users if there were time left over, was the use case from chapter 4.4.3. There were some additional adjustments which had to be made to complete this task in form of a second rule. This turned out to be a challenging task for most users, which was as the thesis workers intended. The tasks before this one had no need of creating additional rules. This lead to most users not trying to click on the green plus button in interface 1 located to the right hand side of the action box. A few users were also confused by the two save buttons, and did not relate to one being for saving a created rule, while the other one was for saving the whole configuration made. These are some of the feedback gathered for improvements needed for future development.

Observations made on the test users for interface 2 showed that there were some confusion about the scenario tab. During the test session, there were no pre-configured scenarios to use. This made the scenario lose its purpose and needed to be skipped by the test users. Most of the testers of interface 2 started by creating a name for their configuration, but got stuck after coming to the scenario tab. They wanted to explore the entire page before continuing on to the next step, while some just skipped right past it after a few seconds.

Interface 2 also had some lack of functionality, which lead to some complications for a few users. They wanted to change a rule they created, but it was not possible to do that in the interface 2 as the feature to do that were missing. It was meant to be possible to do that, but due to lack of time in the development phase, the functionality for removing or changing a created rule were not finished. Therefore, the test users were instructed to verbally tell the observer of when they wanted to remove or change a rule, and how they wanted to apply the changes instead. This affected the time of the configuration in a positive way for the user, which was in the end negligible.

# Chapter 5

# Conclusions

This chapter showcases the conclusion of the whole thesis. The conclusion of the questions at issue, as well as the result of the final interfaces are presented and discussed in this chapter.

## 5.1   Conclusion of Questions at Issue

**How will the use cases be elicited?** The thesis workers started off by holding a brainstorming session with the main goal of finding scenarios which could be translated in to use cases later on into light. The scenarios were then separated into different categories and use cases were made from scenarios without the categories overlapping. The use cases were then elicited by a checklist to see if different possible areas were covered in the scenarios. The use cases were then reconstructed and finalized in iterations.

**Which interface is the most time efficient?** When all things are considered regarding time based efficiency as well as overall relative efficiency, the numbers shows that interface 1 is more accurate when looking at successfully configured systems, while it is more time efficient. As shown in figure 4.18 interface 1 is 43% more time efficient than interface 2 and with the consideration of successfully finished tests it's 500% more efficient. When taking the relative time efficiency in consideration it's only 283% more efficient for the successfully finished tests, as seen in figure 4.19.

**Which interface is compatible to most use cases?** When looking at the effectiveness of the interfaces, there is a clear winner of which interface is the most compatible to most use cases. Figure 4.14 shows the result of successful configurations made, and the graph shows that interface 1 has a higher success rate in all the tasks tested.

When taking satisfaction into consideration, an average SUS score of 53 vs. 58 showed that interface 2 had marginally better results than interface 1.

The effectiveness had significantly greater difference in the result and are therefor of greater importance. But it doesn't mean satisfaction is something that should be completely left out. If the result of the satisfaction measurement would vary more, it would have a greater importance for this conclusion.

According to effectiveness, efficiency and satisfaction it felt safe to draw the conclusion of interface 1 being the most compatible interface to most use cases.

**How will the configuration of events be saved?** Through the whole thesis, discussions about how event configuration done in the interfaces will be saved have taken place. The participants in the interviews stated their preferences in this subject to be as templates. This has been taken into consideration and the analysis of the workshop has shown that this is a good option. Therefore, in both interfaces, there are possibilities of saving a made configuration as a template which will be saved globally. This means that it is available for everyone using the system. The user can at a later time open the saved template, to make further configurations of it and save it under a different name to save it for each site, under a name for what it will actually be used for.

## 5.2    Final Conclusion

The authors liked the flow of how the thesis has gone, but would like to have a bit more time in the development phase for the interfaces. It would also be interesting to speculate in how the results would have differed if there were more test subjects available. There were discussions among the thesis workers of how the students affected the test results in comparison to how it would have been if all test subjects were all experienced AVHS users.

Given the results, interface 1 were expected to have a higher usability score of the two interfaces which was not the case. The success rate of interface 1 was higher than interface 2, but the usability score went in favor of interface 2. This can be related to the feedback given for interface 1 in chapter 4.6.3 from the test sessions where tooltips or short descriptions were mentioned. If the test users were given some kind of help to make the transition from the simple view to the advanced view where configurations were made, the satisfaction of interface 1 might have been different to what it was. Another aspect of what might have changed the satisfaction outcome is with the help of some tooltips or short descriptions on how to navigate in and manage interface 1. The advanced view in interface 1 were designed to show all options available in one screen, which could have made the test users overwhelmed. This could have been prevented by having short descriptions and tooltips to help the user through a configuration.

The feedback given to The reason to why the authors expected interface 1 to have a higher score in usability was due to it being designed in a way that the users had all the options possible the same view. Interface 2 was designed as a step by step kind of wizard and the overview of where the user were located at in the configuration were not as clear.

When the authors were testing the two interfaces, their personal thoughts were that configuration of the use cases were easier to make in interface 1. These results lead to the expectations of interface 1 getting a higher usability score. The reason to why that was not

the case could be related to the authors having the opinion of interface 1 having a faster learning curve than interface 2 after observing the test users. Interface 1 also had a higher success rate for all the tasks tested as well as a shorter completion time for all the tasks.

The thesis workers would therefore like to draw the conclusion of interface 1 to be the most appropriate interface of the two for event configuration between relations of triggers and actions.

## 5.3 Ethical Reflections

The usage of video surveillance is growing as the time goes and it seems like it will continue to grow in modern society [1]. Although monitoring is useful it might not always be welcomed where it is set up. Video surveillance can be interpreted as a violation of privacy of an individual if the surveillance is made incorrectly. That is why all rules and laws has to be followed strictly to justify the surveillance. Therefore, it is very important to think through if video surveillance is really necessary, and what purpose it will fulfill [8]. Depending on the answer, maybe the problem can be solved in a different way.

There are three different aspects to keep in mind when considering if video surveillance is necessary.

**The size of the area -** The bigger the area being under surveillance is, the more people are affected by it, which means that a violation of privacy is being made to a larger group of individuals.

**The vulnerability of the area -** The more vulnerable the area being under surveillance is, the more the people would like to protect it. Having security cameras in for example bathrooms or dressing rooms can be very sensitive. People could be offended by knowing that surveillance is being made in areas similar to these.

**The time of the surveillance -** The longer people are being under surveillance, the more they get offended by it. By watching over people for a longer time, the more it becomes a violation of their privacy and people might not really be comfortable of being watched.

Even if people are experiencing being under surveillance to be a violation of their privacy, it is in some situations necessary. Some of the reasons to justify the monitoring is to prevent crimes from happening, using the gathered data as evidence as well as to protect employees in companies and make civilians feel safer by monitoring specific areas.

## 5.4 Future Work

In this chapter, features or functions which were desired from Axis side, but not within the range of the scope for the thesis are presented in a list. The list is a suggestion to what might be of importance for future work.

### 5.4.1 Documentation for installation

The maintenance of the system as well as a way to inform (like a clear report) when a maintenance or an installation has been made.

How can the person in charge of setting up the system account for that the system has been installed correctly in detail? It is hard to get an overview of what one has set up with the different features like alarm zones, schedules, and other things. Some kind of page for summarizing the setup made would be really desirable.

And if there were some kind of support from the interface to generate such a report it could be a solution to this wish. This feature was outside of the scope of the thesis and was not implemented.

### 5.4.2 Speed and Simplicity

Axis aims to keep their services fast and simple to save both time and money, as well as frustration loading times. This gives their customers a more professional feeling to the system. The implemented interfaces for this thesis are not optimized for great loading times. They had to be implemented in a short timespan and because of that, there wasn't any room to implement speed optimization techniques. They are good enough though as prototypes but far from being ultra fast.

### 5.4.3 Account Security Levels

To be able to control and configure a normal user's account directly from an administrators account without the need of knowing the normal user's login credentials. This is a wish gathered from the interviews in the beginning of this thesis. This thesis didn't include this kind of functionality, since it only focuses on the interfaces for event configuration itself.

### 5.4.4 Interface improvements

According to the result, both interface 1 and interface 2 had a SUS score around the OK threshold, which means they could be improved a lot. By adding the following functionality, the SUS score could be raised.

**For interface 1:**

- Add some kind of description, like a tooltip when hovering on different kinds of objects in the interface.

- Make the drag and drop items look draggable in interface 1. Make it look more consistent in the design.

- Complete the template manager functionality.

- Add the functionality to remove a preset.

- Change the + and - buttons to the right side of a rule to *Add new rule* and *Remove rule* buttons.

**For interface 2:**

- Finish the missing critical functions in interface 2 like edit and delete rule.

- Rename scenario tab in interface 2 to template manager instead. Like in interface 1

- When selecting triggers, change the way available triggers shows. Don't remove the unavailable triggers, disable them instead.

- Add some kind of description, like a tooltip when hovering on different kinds of objects in the interface.

- Improve the design to be more self explanatory and cut down the current text description.

# Bibliography

[1] Andreas Björklund, Kenneth Ulander. Ökning av övervakningskameror. `https://www.svt.se/nyheter/inrikes/stor-okning-av-overvakningskameror` (visited on 11/08/2017).

[2] Axis Communications. About axis communications. `https://www.axis.com/global/en/about-axis` (visited on 24/04/2017).

[3] Axis Communications. About axis communications. `https://www.axis.com/25years/files/axis_history.pdf` (visited on 24/04/2017).

[4] Balsamiq. About balsamiq. `https://balsamiq.com/company` (visited on 07/04/2017).

[5] Bootstrap. About bootstrap. `http://getbootstrap.com/about/` (visited on 05/08/2017).

[6] Jerry Cao. *The Guide to Mockups.* UX Pin, 2016.

[7] Martin Christensen. Modern ux: Design studio. `http://modernux.se/docs/designstudio/` (visited on 17/05/2017).

[8] Datainspektionen. Kameraövervakning på arbetsplatser. `http://www.datainspektionen.se/lagar-och-regler/kameraovervakningslagen/kameraovervakning-pa-arbetsplatser/` (visited on 13/08/2017).

[9] @fabriciot and @caioab. Ux design methods. `https://uxdesign.cc/ux-design-methods-deliverables-657f54ce3c7d` (visited on 24/04/2017).

[10] Jeff Sauro. Sus measuringu. `https://measuringu.com/sus/` (visited on 25/06/2017).

[11] Jeff Sauro. Sus positive/negative. `https://measuringu.com/positive-negative/` (visited on 25/06/2017).

[12] Yvonne Rogers Jenny Preece, Helen Sharp. *Interaction Design - Beyond human-computer interaction*. John-Wiley-and-Sons-Ltd, 2015.

[13] jQuery. jquery - what is jquery. `https://jquery.com/` (visited on 05/08/2017).

[14] JUS. Journal of usability studies. `http://uxpajournal.org/wp-content/uploads/pdf/JUS_Bangor_May2009.pdf` (visited on 25/06/2017).

[15] Justin Mifsud. How to use the system usability scale (sus) to evaluate the usability of your website. `http://usabilitygeek.com/usability-metrics-a-guide-to-quantify-system-usability/` (visited on 01/08/2017).

[16] Soren Lauesen. *Software Requirements: Styles and Techniques*. Addison-Wesley, 2002.

[17] Kevin Matz. *Designing Usable Apps - An agile approach to User Experience Design*. Winchelsea-Press, 2013.

[18] Nigel Bevan. Iso and industry standards for user centred design. `http://usabilitynet.org/trump/documents/Usability_standards.ppt.pdf` (visited on 01/08/2017).

[19] Peter-Paul Koch. Javascript - general introduction. `https://www.quirksmode.org/js/intro.html` (visited on 05/08/2017).

[20] The PHP Group. What can php do. `http://php.net/manual/en/intro-whatcando.php` (visited on 05/08/2017).

[21] Usability.gov. System usability scale. `https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html` (visited on 25/06/2017).

[22] W3Schools. Css tutorial. `https://www.w3schools.com/css/default.asp` (visited on 05/08/2017).

[23] W3Schools. Html introduction. `https://www.w3schools.com/html/html_intro.asp` (visited on 05/08/2017).

[24] W3Schools. Javascript versions. `https://www.w3schools.com/js/js_versions.asp` (visited on 05/08/2017).

# Appendices

# Appendix A
# Interviews

# A.1   UX Advisor

**Background**

- **What department are you working in?**

AVHS - Working in the groups for both new business and UX community.

- **What assignment do you have in your department?**

Works there as a role named UX advisor. This gives her the right to work in other projects as well as her own group project simultaneously. The objective is to find different UX on AXIS. Does not work 100% in AVHS. Wants to do more user research. Mostly strategy, planning, design, flows and sometimes graphic design and user tests.

- **How often are you in contact with the current software? (AVHS)?**
  - and how?

Little with AVHS every day. Two main assignments:

- Feature board, all kind of preparations are done for the people that will be in contact with these features. Finding good features for AVHS? Can it be described in some kind of flowchart? Find hosting providers? Or simply try to find a flowchart for something that is on a wishlist for a customer but the company doesn't really know how to realise it yet.

**Main questions**

- **The previous use cases described in previous meeting were interesting for us, can we go through them again?**

Commissioner:
 Main assignment:
After the contract has been made for which cameras are to be purchased, and also where their positions should be. The main job of the commissioner is to get a customer started. There are some steps to be taken to get there though:

- The cameras are sent to the alarm company and they write down the MAC-addresses for the cameras to the customer to assign them to that company.
- They Create a user for each customer
- They know which cameras are to be assigned to the customer
- They know where the cameras are to be positioned
- They know some of the configurations the cameras will be set up to(Picture quality)
- They know the configurations for how the sensor will behave
- The zones for a site is set up
- What will the triggers be?

Installer:
- Sets up the network connection
- Fixes the cable work
- Connects the cameras to the network

When all is finished and done, they call the alarm company which is the commissioner to notify them of that the system is ready to run.

When the installer has called the commissioner that the system is ready the commissioner runs some kind of a quick test of the system to see that that all the sensors are functional. When all of it looks good, they start off by putting up a schedule for the site(a specific area), when will the sensors be activated inside the store, and when will they be activated for outside the store. They also test the system, like they arm the system and then walks where the alarm should trigger to see if it triggers.

New feature - Arm <-> Disarm
Works in the way that the alarm is always active, but in the different parts of an area, i e in a house, when you are on the 1st floor, the 2nd floor is alarmed, when you go to the 2nd floor, the 1st floor gets alarmed. It's not seen as "on" or "off" but instead seen as different states, "this area armed" or "another area armed"

- **How do you formulate a use case?**

She is basing her use cases from examples for a specific user-group. The problem is that the use case isn't fully testable because a tester reads it very strictly, and they should. So it's not a very good way to actually formulate a use case. But it's a start in the process. She usually tries to formulate a feature and start from there.

For example, there is a user and he has two sites with two cameras for each site. Something happens (ex. the camera was disconnected), there is a signal sent to AVHS and there is some rules to address what and how something should be notified. For example an error to the monitoring station and then they may send a guardian to investigate if there's some problem on the site (physically). And in the same time a mail was sent to someone responsible for the system. And depending on the problem they may need to send out a technician to fix the problem.

Now when we have this steps she tries to write down exactly what's happening in every step.
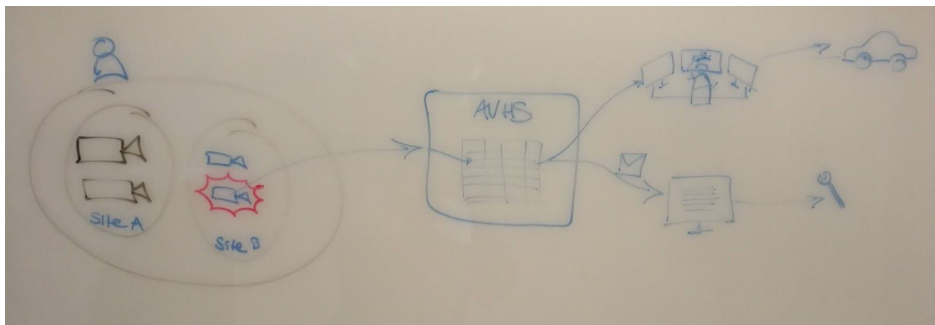


- **What's the ideally system? what features would you like to have? (The ideally AVHS)**

64

The commissioner should be able to login with it's own credentials and then be able to configure the normal users configuration from its own account. Without logging in to the system with the normal user's credentials. The commissioner should also be able to see whatever he needs to see in the same dashboard.

The main purpose is to create a uniform interface with different parts (apps) that's connected through a main menu.

The system should have an interface where multiple roles has different options. The commissioner should see something more advanced and the store manager should be able to see some more basic options and features. The current "Live view" page should be a different application for the end user. And in the top menu the list of apps should be presented for different roles. For example the starter has a bunch of options, the store manager has a few basic options and the store staff has only the live view option.

To configure multiple devices at once she didn't have a specific answer but she gave some examples. Maybe you would like to have a "wizard", or maybe you can copy a configuration to another device, or maybe an entire app for the configuration.

We (AVHS) needs to be able to configure multiple devices faster than today.

- **How many sensors do you think is a realistic scope to be aiming at? 10-100 sensors?**

She describes that 2-16 devices are the "sweet spot" for how many devices a company wants. It is realistic to build a system for 2-4(2-6) devices, or at least optimize it for amount of devices with the possibility to configure more devices(up to 16).

- **Why start the configuration process with the trigger part and your thoughts about our draft?**

She doesn't want to be forced to use one action for a specific trigger. She wants to be able to configure multiple actions for one trigger. And that's how she's been thinking for a while, so it's hard to change perspective. The most commonly used actions for a trigger are record and notify. It's hard to make a simple interface to configure an advanced feature.

Maybe you should create a simple trigger for multiple actions. For example multiple cameras triggers when their own VMD (Video Motion Detection) notifies something. She thinks this way is the most important one, simple configuration for multiple devices rather than advanced configurations for a few.

There are many reasons why the settings are as they are, in the meaning of that you can only edit one camera at the time. Every camera needs specific settings because they may

be looking at something in the picture and not listening to a specific sensor. Depending on what specifically it is looking for, the settings might be quite advanced.

All cameras don't have the same sensors as each other, maybe only one or two of the purchased devices has PIR sensors, which means that the camera with specifically that sensor has to be configured separately, different from the other devices. Another example is that when there is a panic button, it might only be connected to one camera, so when you press the panic button, only that specific camera is triggered.

There are requests for the zone configuration to handle the configuration in a more manageable way, as it is now the one configuring the system, which is the commissioner - has to jump between many interfaces to do all the configuration needed which can take a lot of time and effort. We agreed on that it is not optimal as it is now.

**Stream profiles -**
Different devices has different channels because the cameras has been given different chips. Some cameras can stream 4 channels, and one of these are always reserved for the highest resolution. There are default stream profiles for all cameras that gets configured when you connect the camera for the first time and they are available on the interface and the customer can change the setting. Axis do not want to have this visible because the settings should be consistent depending on the subscription that you have signed. It also has to do with which quality you need for where the data is to be saved. Medium can mean different settings for different devices and subscriptions.

If a camera is able to stream from 4 channels, which 1 is set up to the highest resolution, you are left with 3. If you put one of these to take thumbnail photos, one to continuously record and one to see the stream live at all times. If you then want to have another stream to record live you are left with no channels and do not get a picture for that setting, which is a big problem. The current solution to this is that there is something called stream profiles which lets you set up as many channels as your camera allows you to.

This can be 4 streams on some devices, and there will be some devices with 2 in the future, or more than that on some devices. If you choose the quality on the device, there is some kind of setup which makes it duplicate the stream in the same quality to somehow get more channels that it is set up to be. Depending on the max resolution for the cameras, this feature gets limited somehow as well.

AXIS wants to make the configuration in general easier than it is today. As it is now it is already too hard for the customer to do the configuration. Some devices doesnt show on the interface, and they want to put up a device list on the front page. They would maybe rather do easy configuration on many devices instead of complicated configurations on many devices.

# A.2   Software Engineer

**Background**

- **What department are you working in?**

Working in the department of AVHS.

- **What assignment do you have in your department?**

Software engineer and acting as a part of the support team

- **How often are you in contact with the current software? (AVHS)?**

All the time, every part of the system

**Main questions**

- **Which use cases are useful to you, which use cases do you wish for us to take a look at?**

Two things:
Groups - When you are configuring multiple cameras, you should be able to configure multiple cameras at the same time. It's unreasonable to configure every single camera one by one with the exact same settings.

Trigger for multiple conditions - combine multiple sensors with a condition.
Maybe if the action is the same for the entire group but the trigger could be for just one camera.

For example:
If you have a bus, there may be around four cameras. If they are placed from the front to the back facing the back. If the first camera detects sound. The sound is right below the first camera (in the front). Then the sound may not be equally loud in the back. So let's say it's only the first camera who is triggering on the sound level condition but then every camera on the buss should start record video anyway, even if their sound condition isn't fulfilled.

- **What is your opinion about reaching a finished configuration in a more advanced way but on few clicks vs an easier configuration with more clicks? Maybe you could use a standard template?**

Templates are good if they work. But sometimes they doesn't match the user needs. 90% did match but not the rest 10% of what the user needed. It's easy to be stuck in a loop trying to find the right template instead of actually start to configure the device. It may be a good idea if you are able to choose a template close to the user needs and then reconfigure the template for the exact purpose.

It will always be more work to configure the configuration if there are more features. It's like a quota. If you have one unit work and one feature then we have a 1:1 ratio. If you add more features you also add more work.

67

- **How would you describe an ideal system vs a realistic system(AVHS) ?**

Like I said before, groups and relation between multiple triggers. Like the buss example.

- **What's your priority and why? Configure multiple simple settings for almost every device and then configure the advanced settings on the few remaining devices. Or configure everything in the same interface, both advanced and simple settings at the same time.**

I would like to choose a trigger and then the action for some cameras. For example: When there is a motion and a sound detected I would like to record video for camera 1, camera 2 and camera 3. Then I'm done.

Let's use a buss example again. If there is a fourth camera facing the bus driver it may not trigger for the motion detection because he is moving all the time. But it still needs to record when the rest of the group starts to record (cam 1,2,3). This could be a use case scenario.

- **Other thoughts, if you have some specific goals or if you have something you wants us to look at.**

There are two features I want. It's logic and it's grouping.

# A.3  Software Engineer

**Background**

- **What department are you working in?**

Working in the department of AVHS.

- **What assignment do you have in your department?**

Software engineer and acting as a part of support/customer team team

- **How often are you in contact with the current software? (AVHS)?**

Working with the current software of AVHS every day in the way of troubleshooting the system and handling bug fixes. Also works with feature development for the future system.

**Main questions**

- **What is requested of the system(configuration) by the clients on the market?**

The person in the interview thinks that simplicity is the most requested thing in the configuration system from the customers, and also quick systems. He thinks that the settings will be set few times, in example on the installation and then it is supposed to work without problems until something breaks. Once you set up the system, there is no more need to get into the configuration.

- **Do you think there is a need from customer that the devices can be triggered in relation to each other?**

Even if the customers aren't aware of it, there are a lot of things that could be done with the function of relations between devices. There are a lot of services that can be sold with this feature.

An example is when someone uses their keycard, you want to identify the person using it to compare if the keycard belongs to the person using it. If not, that's a trigger for a desired action in that specific case.

The person in question thinks it is very important to identify the services that can be useful for the market rather than features able to be programmed, since generally that is what happens - you sell a service and then you want to be able to configure the service to the customers needs.

50% - 80% of the customers devices will be configured in the same way, which means that this is the area that needs to be covered with basic configuration settings.

- **Do you think there is a desire from the customers to be able to configure multiple devices simultaneously?**

If there is a possibility to save them time and if it adds simplicity, the person thinks that there is a desire to be able to configure multiple devices simultaneously.

If the configuration of the system goes quickly, the company saves both time and money.

- **Which use cases are useful to you, which use cases do you wish for us to take a look at?**

There are generally two modes in an alarm system, On and Off. But off doesn't generally have to mean that the system is not on. It might just not give you a notification when you do not want it.

Example is an alarm system for a home, you might have specific settings for the system when you are at home, and when you are out. You don't want to adjust all the settings when you get home to the settings you want when you are home. You just want to be able to set those things once, and when you are at home you press the button "home" for those settings, and when you leave, you just press "away" for those settings to activate. Maybe some additional modes should be programmable on the devices, like "vacation"?

Another interesting view given by the person is that you could possibly turn on the conditions, like if there is someone outside the door and tries to open the door - and the door gets unlocked within 2 minutes, that is okay. But if someone tries to open the door, and the door doesn't get unlocked within two minutes, that might be something fishy going on.

This scenario can be used in different ways, for example if you have kids and your kids usually comes home at a specific time, if they have not unlocked the door and entered their home by that time, is there something strange going on or what is the reason for the delay?

The important thing we can get out of this is - when should the information be delivered to the person? Is it possible to turn it, can we flag for when something that should have happened, didn't happen?

- **What is your opinion about reaching a finished configuration in a more advanced way but on few clicks vs an easier configuration with more clicks?**

It all depends on who would manage the system. And are the settings similar to each other? If yes, maybe save the setting and apply the setting done 10 minutes ago on chosen devices.

Maybe you want to edit devices, maybe many simultaneously or edit one, the setting gets saved as some kind of template and you can apply it on many devices.

It might be desirable to configure many devices at the same time with simple settings and then configure each device individually for the ones that needs more advanced configuration.

- **How would you describe an ideal system vs a realistic system(AVHS) ?**

What is missing today is an easy way to create relations between sensors. Today, it is possible but the problem is that it is very sensitive and very tricky to do it, and there is no way of seeing which devices has a relation.

Also, the ability of the system to notify the user of when something happens is not really optimal. The user gets very little information about when something happens or what happens. The person asked thinks that the recording time in the future will increase since internet will be more and more available as well as space(harddisks and so on) will be cheaper. This leads to constant recording and the tricky part will be for the system to give information desired  - When do you want a notification and of what?

Many people will assume to start from a **scenario**, and a service package sold - and work from there to how to handle things. This means that you don't start with the most advanced settings, you just start with the settings of the package sold and thereafter configure the additional settings.

Another thing that was talked about a lot is **templates**, is it possible to save a setting given a specific name that was made before and apply it on other devices that needs similar settings? For example "entrance" for stores, or  the recording of the "register machine".

Interesting input for bigger systems were the ability to group devices, and that a device might be active in more than one group, for example for floor 2 AND the staircase for floor 2.

# A.4   AVHS Product Manager

**Background**

- **What department are you working in?**

Working in the department of New business

- **What assignment do you have in your department?**

The person is in charge of the AVHS product management, he is product manager for
AVHS.

- **How often are you in contact with the current software? (AVHS)?**

He is in contact with the software multiple times a week, in form of testing the software,
demonstrating AVHS for potential customers, and also active in areas where there is room
for improvement.

**Main questions**

- **What is requested of the system(configuration) by the clients on the market?**

The person pointed out that this is a very broad question, but we pointed out the importance
of relations and configurations between devices. One of the things requested of the
customers today is the possibility to record on multiple cameras on the same trigger, and this
is important since it will give the viewer a better understanding of the whole incident. Today
there is only one device able to record which means that the viewer is only able to see a part
of the incident which makes it harder to understand the whole picture of how the incident
developed. This is possible with today's system but it is very very tricky and very fragile, it
might not always work.

What is of importance for the person interviewed is the new I/O device with different ports,
where you connect the port of the I/O device to some sensor that could lead to a specific
action depending on the trigger. The I/O device can also work as a trigger for an action. This
can be triggers/actions for different sites or alarm systems.

The person also saw an importance of making the configuration in a way that its possible to
get an overview of what shall be installed or has been installed, this so that it can be
possible to get a clear view of it.

- **Which use cases are useful to you, which use cases do you wish for us to take
  a look at?**

The use possible use cases is how to make the configuration possible, and another use
case is if you want to replace a device, how do you add that to an existing system? The
devices are all unique with MAC addresses.

- **Do you think there is a need from customer that the devices can be triggered in relation to each other?**

There is a big request to make this possible, and if it becomes possible to trigger devices in relation to each other, it will make the product more unique and desirable.

- **Do you think there is a desire from the customers to be able to configure multiple devices simultaneously?**

This is also a request, to be able to configure devices on many devices simultaneously with similar settings, and then configure the devices with additional settings after the easy setup is done on them.

There was a request to configure most of the devices together with possibly some kind of wizard or walkthrough.

- **What is your opinion about reaching a finished configuration in a more advanced way but on few clicks vs an easier configuration with more clicks?**

The most important thing for the person is simplicity since the person setting up the system is easily replaceable and the new person might not have the knowledge of the configuration process. There were also a desire of templates, in the meaning of adjusting known devices to the installer - the installer can, if there already exists a template for that specific device just choose that one and then configure the settings or just save them as they are.

- **What's your priority and why? Configure multiple simple settings for almost every device and then configure the advanced settings on the few remaining devices. Or configure everything in the same interface, both advanced and simple settings at the same time.**

it can be complicated to make many configurations on many devices, but it can be better for the user to configure one setting on many devices. There is of greater importance to find a use case which is very normal or returning and then make a configuration to make that configuration process simplified.

The headline is - find something that is very likely to be setup all the time and make that process as easy as possible, and let the advanced settings be set up individually for each device. The ratio for this is < simplicity 80% - 20% advanced settings for expert user > .

There is also interesting to get some kind of measurable input on how the configuration is being run, like how many percent of the user's runs the wizard, or how many of the total user base uses the advanced configuration mode?

- **How would you describe an ideal system vs a realistic system(AVHS) ?**

The relations between devices are seen easily in some kind of overview of the system which shows which devices are configured to work with each other, and in which way? This together with the maintenance of the system and also some way to inform(like a clear report) when a maintenance or an installation has been made.

How can the person which has set up the system account for that the system has been set up correctly in detail? (pdf or something else). It is hard to get an overview of what one has set up with the different features like alarm zones, schedules, and other things. Some kind of page for summarizing the setup made would be really desirable.

- **The summary of the setup order on how to configure the system using our example mockups.**

(P1) I want to do something -> I want to do it on these devices -> When this happens
(P2) Something happens -> I want to do something -> I want to do it on these devices

- **Which order do you think is the most natural way for the user to think? How would you think?**

If you ask a customer, you have a system. What do you want it to do? How does the customer respond? Is it I want to do something when this happens (P1) or is it; When this happens, I want to do something (P2).

His spontaneously opinion is when something happens, do this for these devices. (P2)

**Gave a scenario as an example:**
When you get to a customer and asks; What do you have here?
  - He has a shop, or maybe an outdoor area.
Do you want to protect it, and why?
  - I have a problem, people are jumping over the gate.
Okey, so you want to do something when people jumps over the gate? What do you want to do?
  - I want to electrify the gate and record some video of the event.

This may be a good real life scenario and it's based on P2s configuration order. When this happens, do this..

The main question; Why did you buy this device?
There is usually a good reason in the first place. So why not configure it from the same way?

- **How do you think a use-case should be like? What's a relevant use-case?**

Find real life examples, and how can AVHS help a customer with his issues? How to solve this issue?

Start with a real person/user, then continue with the installer, how should he mount/install it in a simple way?

Usually the person who is installing a camera is good with cables and the physical installation process. Then there is another person configuring the camera and that's his area of expertise. It's important to keep the entire process as simple and fast as possible.

74

- **Is there something else you want us to look into? Or do you have another input?**

It's very common to get a phonecall from a customer who would like to do an action on a few more cameras if something triggers the first one. It's a very common issue today for AVHS. The normal response is usually; Yes, you can do it, but you would probably not want to do it.. Because it's complicated to setup and it won't work in the future because it needs a lot of maintenance later on.

It's also common to have the same configuration on multiple devices at the same time.

Try to think outside the box.

And remember the new I/O Box.

# Appendix B
# Scenarios

## Scenarion som kan vara intressanta

1. **Inbrottsskydd**
2. **Tillträde till lokal**
3. **Incident på allmän plats**
4. **Guard tour**
5. **Installations övervakning**

### 1. Inbrottsskydd

*Detta är intressant för t.ex en butik under stängningstider eller när det ej är tillåtet för
människor att vara i lokalen. Kan även detektera problem som varför är larmet ej påslaget
när det SKA vara påslaget, eller varför är larmet PÅSLAGET när det ej ska vara påslaget?*

Notifiering vid folk i lokal under stängningstid
Video/bild vid folk i lokal för relevanta enheter
Händelser vid folk i lokal t.ex uppbruten dörr
Larma om skyddet EJ är igång under stängningstider
Larma om det ska vara larmat under en tid men det ej är det (Glömt slå på larm)

### 2. Tillträde till lokal/område

*Detta är intressant för lokaler som har någon typ av igenkänning mekanik av människor som
vill komma in i lokalen. Det är även intressant för de som ej vill att någon ska befinna sig i
närheten av lokalen under en viss tid, eller överhuvudtaget.*

Fel person kommer/försöker komma in
Tailgating    - någon går in tillsammans med någon som har tillgång till lokal
Tampering    - någon försöker komma in i lokal efter att ha manipulerat nyckeln/låset
loitering    - någon befinner sig på en plats som ej har tillträde till den platsen

### 3. Incident på allmän plats

*Detta är intressant för någon som vill veta om det sker något som är ovanligt, eller något
utöver det vanliga på allmänna platser. Varför registrerades ett skrik, varför var det en
ljudökning, varför registrerades hastiga rörelser, eller varför befinner sig ett objekt på en
plats under en längre tid?*

Larma vid incident
Reagera på skrik / hastiga rörelser
Loitering

## 4. Guard Tour

*Detta är intressant för företag som har vakter som går omkring i lokalen under speciella tider. Händelser som kan vara intressanta att notera är saker som hur lång tid tar det för vakten att gå sin runda, varför är inte vakten där han/hon bör vara vid en tidpunkt eller varför går inte vakten sin runda alls?*

Vakter som inte gått sin runda ska generera larm
Slå av motion detection (för inbrottslarmet) i t.ex. 10 minuter medan vakten går sin runda

## 5. Installations övervakning

*Detta är intressant när något statiskt har förändrats. När statiska platser helt plötsligt ser ut på något annat sätt så sker t.ex. ett larm. Kan även användas med ett statistiskt syfte.*

Larma när något har förändrats
Använda med statistiskt syfte
- Graffiti
- Museum
- Skyltfönster
- Butikshyllor

# Appendix C
# Interfaces

# C.1 Interface 1 - Advanced view

# C.2   Interface 2 - Overview

# Appendix D
# Test-data

# Interface 1

| Role | Engineer | Software Consultant | Software Developer | Student worker | Test Consultant | Summer Worker | Exp Engineer |
|---|---|---|---|---|---|---|---|
| Role category | Employee | Employee | Employee | Student | Employee | Student | Employee |
| 1. I think that I would like to use this product frequently. | 4 | 2 | 4 | 3 | 4 | 2 | 2 |
| 2. I found the product unnecessarily complex. | 3 | 3 | 2 | 3 | 3 | 4 | 3 |
| 3. I thought the product was easy to use. | 4 | 3 | 4 | 3 | 3 | 1 | 3 |
| 4. I think that I would need the support of a technical person to be able to use this product. | 1 | 4 | 3 | 4 | 2 | 4 | 4 |
| 5. I found the various functions in the product were well integrated. | 4 | 2 | 4 | 3 | 3 | 2 | 3 |
| 6. I thought there was too much inconsistency in this product. | 4 | 3 | 2 | 2 | 4 | 1 | 2 |
| 7. I imagine that most people would learn to use this product very quickly. | 4 | 2 | 4 | 3 | 4 | 3 | 4 |
| 8. I found the product very awkward to use. | 3 | 4 | 2 | 3 | 2 | 3 | 1 |
| 9. I felt very confident using the product. | 3 | 3 | 3 | 2 | 3 | 1 | 3 |
| 10. I needed to learn a lot of things before I could get going with this product. | 2 | 4 | 3 | 3 | 1 | 4 | 1 |
| SUS Score | 65 | 35 | 67,5 | 47,5 | 62,5 | 32,5 | 60 |
| Clicks | 158 | 249 | 183 | 147 | 345 | 183 | 205 |
| Time | 600 | 1200 | 540 | 1200 | 720 | 720 | 800 |
| Time incl. bonus UC | 930 | 1446 | 841 | 1206 | 1019 | 932 | 952 |
| Numbers of UCs completed | 4 | 3 | 4 | 3 | 4 | 4 | 4 |
| Within timeframe | Yes | Yes | Yes | Yes | Yes | Yes | Yes |
| Successfully completed bonus UC within timeframe | No | No | Yes | No | Yes | No | No |

# Interface 2

| Role | Exp Engineer | Student | Student tester | Exp Engineer | Exp Engineer | Summer Worker | Engineer |
|---|---|---|---|---|---|---|---|
| Role category | Employee | Student | Student | Employee | Employee | Student | Employee |
| 1. I think that I would like to use this product frequently. | 3 | 2 | 4 | 4 | 3 | 3 | 4 |
| 2. I found the product unnecessarily complex. | 2 | 3 | 3 | 2 | 3 | 2 | 4 |
| 3. I thought the product was easy to use. | 2 | 2 | 2 | 3 | 4 | 3 | 2 |
| 4. I think that I would need the support of a technical person to be able to use this product. | 3 | 2 | 4 | 1 | 2 | 1 | 4 |
| 5. I found the various functions in the product were well integrated. | 4 | 5 | 2 | 3 | 4 | 4 | 4 |
| 6. I thought there was too much inconsistency in this product. | 2 | 3 | 1 | 2 | 1 | 2 | 2 |
| 7. I imagine that most people would learn to use this product very quickly. | 3 | 3 | 4 | 4 | 4 | 3 | 2 |
| 8. I found the product very awkward to use. | 4 | 2 | 2 | 2 | 3 | 3 | 2 |
| 9. I felt very confident using the product. | 2 | 2 | 2 | 3 | 4 | 2 | 2 |
| 10. I needed to learn a lot of things before I could get going with this product. | 3 | 2 | 4 | 2 | 2 | 3 | 1 |
| SUS Score | 50 | 55 | 50 | 70 | 70 | 60 | 52,5 |
| Clicks | 226 | 258 | 144 | 183 | 204 | 290 | 277 |
| Time | 973 | 976 | 1194 | 1102 | 898 | 1124 | 1142 |
| Time incl. bonus UC | 973 | 976 | 1194 | 1102 | 898 | 1328 | 1337 |
| Numbers of UCs completed | 3 | 3 | 2 | 3 | 4 | 4 | 4 |
| Within timeframe | Yes | Yes | No | Yes | Yes | Yes | Yes |
| Successfully completed bonus UC within timeframe | No | No | No | No | Yes | No | No |