



CHALMERS
UNIVERSITY OF TECHNOLOGY



Designing a virtual receptionist

Using contextual information and natural interfaces

Master thesis in Interaction Design and Technologies

OLIVER CARLSSON
DANIEL STRÖM

REPORT NO. 2015:132

Designing a virtual receptionist

Using contextual information and natural interfaces

OLIVER CARLSSON

DANIEL STRÖM



CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Interaction Design and Technologies
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2015

Designing a virtual receptionist
Using contextual information and natural interfaces
Oliver Carlsson and Daniel Ström

© OLIVER CARLSSON, DANIEL STRÖM, 2015.

Technical report no 2015:132
Department of Interaction Design and Technologies
Chalmers University of Technology
SE-412 96 Göteborg
Sweden
Telephone + 46 (0)31-772 1000

Examinator: Staffan Björk
Supervisor: Josef Wideström

Cover:

An initial prototype of an organically growing interface suited for large display surfaces.

Chalmers Reproservice
Göteborg, Sweden 2015

Designing a virtual receptionist
Using contextual information and natural interfaces
Oliver Carlsson and Daniel Ström

Department of Interaction Design and Technologies
Chalmers University of Technology

ABSTRACT

In this thesis, we describe our work in designing and building a virtual receptionist system, called “*Ameë*”, for Interaktionsbyrå’s studio in Gothenburg. Ameë’s responsibilities includes welcoming people to the studio and providing services to visitors and employees alike. A significant part of the project concerns how to make the receptionist context-aware and have it interact with the user in a natural human way. To achieve this we built a receptionist system based on a network of microservices that collaborate to create a unique user experience using face recognition and text-to-speech technology.

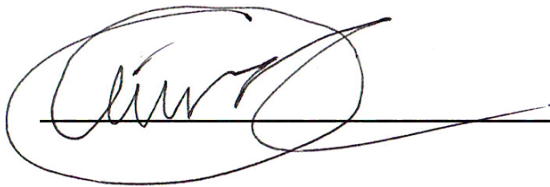
In our work we identify and discuss challenges in building context-aware applications and how publicly available information can be used to enhance the user experience. Going further, we examine how such a system might be received by users through a series of user tests and interviews. Our findings show; that users enjoy being received by Ameë, that they have few concerns about privacy in this context, that further study needs to be done in how to manage the quickly increasing complexity of a conversation model. The project contributes to the current body of research as a cursory investigation into the area of designing automated personal service agents for a non-personal context.

Keywords: Interaction Design, Context aware, Digital privacy, Natural interfaces

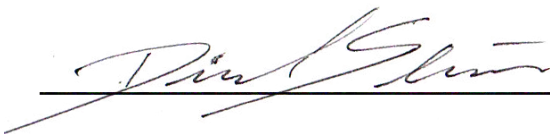
Acknowledgement

The project as it is, could not have been achieved without the support and guidance from a number of amazing people. We, the authors, would like to extend our sincere thanks, in no particular order, to Oskar Hagberg - for keeping us in check and on track; to Josef Wideström - for helping us think outside the box; to all the talented people of Interaktionsbyrån - who've helped, inspired, critiqued and challenged us; and lastly to our friends and family - who've offered fresh insights and a different perspective.

To all of those above, *thank you!*

A handwritten signature in black ink, featuring a large, stylized 'O' and 'C' that loop together, followed by a horizontal line.

Oliver Carlsson

A handwritten signature in black ink, featuring a stylized 'D' and 'S' that loop together, followed by a horizontal line.

Daniel Ström

Contents

1. Introduction	1
1.1. Stakeholders	3
1.2. Amee	3
2. Background	5
2.1. Personal integrity and service	5
2.2. Existing products	8
3. Theory	10
3.1. Identification	10
3.1.1. Facial recognition	10
3.1.2. Mobile sniffing	12
3.1.3. BLE	12
3.1.4. Wifi-sniffing	13
3.2. Ambient Intelligence and Context-Aware Computing	13
3.3. Natural interaction	14
3.4. Physical Space	16
4. Methodology	18
4.1. Theoretical frameworks and methods	18
4.1.1. Action Research	18
4.1.2. On-site development	19
4.1.3. Microservices	20
4.1.4. Rapid Prototyping	20
4.1.5. Continuous Delivery	20
4.1.6. Automated testing	21
4.1.7. Test-driven development	21
4.1.8. 100 ideas in an hour	21
4.1.9. User stories	21
4.1.10. Interviews	22
4.2. Engineering Tools	22
4.2.1. Node.js	22
4.2.2. Brunch	22
4.2.3. Mosca	23
4.2.4. AngularJS	23

4.2.5. Node-RED	23
4.2.6. OpenCV	23
4.2.7. Qt	24
4.2.8. Hardware	24
4.2.9. Git	24
4.2.10. Mocha.js / Chai	24
4.2.11. Preprocessors	25
4.2.12. Ubuntu	25
4.3. Planning	25
5. Process	27
6. Result	33
6.1. Platform	33
6.1.1. amee-mqtt	33
6.1.2. amee-nodered	33
6.1.3. amee-cam	34
6.1.4. amee-doors	34
6.1.5. amee-admin	36
6.1.6. amee-recognizer	36
6.1.7. amee-boombox	36
6.1.8. amee-greeter	36
6.1.9. amee-controller	37
6.1.10. amee-voice	37
6.1.11. amee-info	37
6.1.12. Custom libraries	37
6.2. Prototype	38
7. Discussion	41
7.1. Creating a welcoming atmosphere	41
7.2. Collecting and using contextual information	41
7.3. Users feelings about privacy invasion	42
7.4. Developing for the future	43
7.5. Planning and Process	44
7.6. Ethical issues	45
7.7. Future Work	46
8. Conclusion	48
Appendices	54
A. Definitions	55

B. User Stories	56
C. Interviews	58
C.1. Employees at Interaktionsbyrån	58
C.1.1. Employee 1	58
C.1.2. Employee 2	59
C.1.3. Employee 3	60
C.1.4. first time user 1	61
C.1.5. first time user 2	62
D. User Scenarios	64

1. Introduction

With the creation of the Internet, ICT (Internet and Communications Technology) has become the backbone of social and political structure. However, new issues regarding personal integrity has become evident as more people have brought Internet into their everyday life. The question of data privacy and personal integrity is a moral, legal and political concern. It influences how people, companies and governments can control their information in an environment, where private personal actions and opinions no longer is bounded by our physical proximity. On the other hand, the availability of easily obtainable, cheap and exact data should also be considered an opportunity to provide adaptive and customized services.

This project aims to create a modular framework to collect and exploit contextual information; in order to provide a personalized user experience and service in a receptionist. By making the collected data visible we also hope to enlighten users about the consequences and potential of public data sharing. The receptionist application will be used in the Gothenburg studio of Interaktionsbyrån where it will welcome visitors and provide personalized services to employees. We see this as a fitting environment to carry out the project in, since the workplace is a setting where personal image can be an extra delicate matter - in business relations and among coworkers. We hope this will further enhance the feeling of importance when unexpected information is made visible.

Interaktionsbyrån is a design studio specialized in interaction within the automotive industry. The studio currently has about 30-35 employees and is rapidly expanding. They are also constantly looking for new potential business fields to expand into. Being a growing small business, it is important to stand out to attract clients and employees. "Ameë", the virtual receptionist (see section 1.2.), is a unique and innovative approach to greet clients and employees, showcasing unique ideas both through the medium of the virtual receptionist, e.g. showcasing inhouse projects, and through the receptionist itself. The project is unique in many aspects, hence we arrive at the main research question:

How can we design and build a virtual receptionist that make visitors feel well received?

In order to succeed in answering this question we believe context awareness is a crucial. We define contextual information as any data with the potential to improve and/or personalize a design artifact's user experience. One example of such information might be the last time a person was seen at the studio. Using that extra knowledge one could change the greeting from a general "Welcome!" to the more personal "Long time no see!" when appropriate. This leads us to the first follow up question:

How can we collect contextual information and convey it through the virtual receptionist?

The contextual information, could be public information that is published openly on the internet or information that has directly or indirectly been shared to us. This information should then be conveyed to the user in such a way as to enhance the experience i.e. information can be perceived differently depending on timing, "tone of voice", previous topics, and context which leads us to the third question:

How will users feel about the digital agent knowing their personal information?

People have different thoughts on integrity, the opinion might change depending on the context, service, but especially how the service is conveyed. E.g. Google provides a ticket feature in Gmail; they scan private messages and in return provide a personal service. It is designed in such a way as to not call for attention and especially not highlight the fact that Google scans all the emails. However, If they would send you a message like this:

Hi, we have just scanned through all of your 5 347 emails but have not found any flight tickets today. Don't worry, we will check every email that you may receive in the future!

From this perspective, how will users interpret and feel for the product we design?

Finally, since the prototype developed during this project is intended to live and evolve past the end of the thesis work, it's important that any system(s) built are constructed with this in mind:

How does one build software that accommodates changing requirements?

Over time the needs of Interaktionsbyrå might change, someone new takes over development or they might move to a new office. The architecture of the prototype should allow for changing its behavior in response to these changes

with as little effort required as possible. This ability is not only important after the end of the thesis work, but is also very valuable during the project as requirements are prone to change even during the execution. While we don't consider the above a *research* question to be answered by this thesis, it's resolution is still an important goal of development and measure of the project's success.

1.1. Stakeholders

The approach during the project has considered the five stakeholders. To do a successful thesis work, all stakeholders requirements needs to be considered, negotiated and fulfilled. The stakeholders' requirements have also been the reason for some major decisions during the project. The five stakeholders are:

Chalmers The university that examines the thesis. Expects a well written thesis and for the work to make good use of research methods.

Nordiska Interaktionsbyrå AB The leading interaction design agency within the Scandinavian automotive industry. The virtual receptionist designed during the thesis work will be built and tested at their studio in Gothenburg. Interaktionsbyrå wants a slick virtual receptionist that will impress their clients.

Thesis Workers Besides satisfying all the other stakeholders, we want to do a project that allow us to make use of knowledge learned within the fields of Computer Science, IT, and Interaction design field and develop it further in a company environment.

Customers and Contacts External people with ties to Interaktionsbyrå. They will want to gain access to the studio and get in contact with the right employee with as little effort as possible.

Employees A real receptionist might provide services not only to clients but to the company's employees. During the project we will look at what services we can offer those working at Interaktionsbyrå.

1.2. Amee

To add personality to the application we decided to call it "Amee". Amee originates from Aml (Ambient Intelligence) which could be defined as:

A digital environment that supports people in their daily lives in a nonintrusive way.

(Augusto 2007)

The point was to give it a humanlike name to increase the sensation of a receptionist rather than an application.

2. Background

There is the technical possibility to gather data about everything we do, and process it in a way that could give few people personal gain and power. But for different reasons the effect of this possibility is still discrete. Regarding this project, we have looked into different digital receptionists and smart doorbells that in some way handles personal data to assist users in the lobby. To discuss how smart these products are in terms data handling, and user experience.

2.1. Personal integrity and service

It should come as no surprise to anyone that there's a wealth of information available about each and everyone of us online. While people in modern times have always left a paper-trail, with the widespread adoption of the Internet, the amount of data and ease of collection has increased with several orders of magnitude. Today, it is quite possible to construct a rather good image of an individuals life, and to some extent even predict his or her future actions (Weiss 2009). The ability to do this is extremely beneficial to a number of actors, not the least of which is law enforcement who increasingly and with great success use digital forensics in the pursuit of criminals (Greengard 2012). However, even given a benevolent government and increased security, can complete access to most, if not all, of our data be justified? If not, how do you define what information is public and what is private?

In the United Nation's *Universal Declaration of Human Rights* (1948), article 12 states:

No one shall be subjected to arbitrary interference with his privacy, family, home or correspondence, nor to attacks upon his honor and reputation. Everyone has the right to the protection of the law against such interference or attacks.

Furthermore, article 8 of the *European Convention on Human Rights* (1950) proclaims:

1. Everyone has the right to respect for his private and family life, his home and his correspondence.
2. There shall be no interference by a public authority with the exercise of this right except such as is in accordance with the law and is necessary in a democratic society in the interests of national security, public safety or the economic wellbeing of the country, for the prevention of disorder or crime, for the protection of health or morals, or for the protection of the rights and freedoms of others.

Clearly, people have a basic right to privacy, though the European Convention on Human Rights provides little restrictions on *governmental* interference. Settling on exactly how far that right to privacy extends has proved difficult. In America the law is left vague - that is private which can reasonably be expected to be private (Weiss 2009). As such a conversation in your home is considered private, and subject to legal protection, but not a conversation at Starbucks. Unfortunately these rights are often interpreted in such a way that they do not extend to information online.

Privacy International (2010) record in their extensive examination of privacy policies in European countries many ways in which current laws fail to protect online privacy. For instance, in Ireland tapping a regular phone call requires a court order but listening in on a VoIP call does not. In countries like Bulgaria and Croatia, retrieving any information from a 3rd party, such as Facebook or an Internet Service Provider (ISP), does not require a warrant. Several countries, e.g. Sweden, also provide very limited ability to audit and review the actions of security services, complicating the discussion on the right to privacy.

The great extent to which modern governments surveil their citizens has incurred heated criticisms in recent times. The discussion grew especially pointed after former NSA employee Edward Snowden revealed that the United States indiscriminately monitors the communications of all its citizens as well as many foreign individuals, one notable example being the Chancellor of Germany Angela Merkel (Poitras 2014). However, even without the exceptional surveillance privileges assumed by governments there is still a wealth of public data available to anyone who cares to collect it.

In the harsh business environment of today, having the most correct and timely information can provide a significant business advantage. How to analyze consumer data for insights has even been identified as the key question for businesses in the 21st century (Rosenberger, Nash, and Graham 2009). Indeed many contemporary companies' success can be attributed to their understanding of their target user. Netflix is a prime example of such a company. Feeding

data from millions of users into its recommendations engine Cinematch they have been able to create a tailored user experience that sets them apart from the competition (Rosenberger, Nash, and Graham 2009). In this case, one can say that consumers trade information about movie preferences in return for a better service.

Movie preferences might seem like a fairly harmless piece of information to share but there are plenty of examples where we share more sensitive data, and even with seemingly insignificant data, given enough of it one can start to build a profile of your personality, tastes and activities.

In an attempt to highlight the issue of data privacy German politician Malte Spitz sued his cell phone carrier in order to force them to hand over any data they had collected on him. He received a spreadsheet with 35 831 rows detailing his movements and metadata on calls and texts for the previous six months. The data coupled with some other public data such as tweets and Mr Spitz's official calendar from his webpage has been published on German news site Zeit as an interactive map (Zeit 2011). From the data it can be deduced where he lives, works, when and how he commutes, where he hangs out, when he sleeps, who his acquaintances, friends and family are and much more. It serves as a chilling example of just how complete of a profile one can construct given only a few sources.

People like Mike Spitz, Julian Assange and Edward Snowden obviously care a lot about the right of privacy but how much do people in general care about privacy? A study found that while many people value their data highly, they are generally not willing to pay much to secure it (Acquisti, John, and Loewenstein 2013). Tomas Chamorro-Premuzic goes on to discuss this phenomenon called the "endowment effect".

The implication is that consumers care more about losing than gaining anonymity, so they will generally not go out of their way to recover the privacy they already lost or gave away.

(Chamorro-Premuzic 2014)

Complementing this notion is a study of over 5000 Facebook users. It shows that the users have increasingly taken measures to restrict their public content over the 6 years they were studied (Stutzman, Gross, and Acquisti 2013). However, changes in how Facebook manages privacy, implemented toward the end of the project, halted and in some cases reverted the trend. Furthermore, the authors note that the amount and scope of personal information that the users shared within their network increased over the course of the study and therefore so did disclosures to unintended parties such as Facebook, third-party apps and advertisers.

2.2. Existing products

There are a few products, both aimed for office and private homes, that have a similar purpose of being a digital receptionist. These products are discussed in the aspect of potential advantages and disadvantages for our product, based on the companies' own presentation.

LobbyConnect by Eventboard is as the name suggests a manual check in system for the lobby. The product statement is:

LobbyConnect is the perfect digital receptionist. Greet visitors, get notified when guests arrive, and make a powerful first impression.

Their system is based on a touch based interaction with an iPad, which offers a few features. Visitor Management, which keeps a record of the visitors by allowing them to manually register at their arrival. Employee Notifications, allows the visitor to notify a selected employee by clicking on them in a contact list. Contracts and Agreements, if the company has an NDA policy this can be signed and stored on the product. (Eventboard 2015)

This product provides the fundamental functions that you would expect from a check-in machine. It can work by an already existing front desk to relieve some workload of the receptionists, or it could work all by itself. The overall interface design is minimalistic and intuitive with nice animated transition. However, it is debatable whether it provides a "powerful first impression", as this system feels rather generic. It is hard to be impressed by an iPad in a lobby when it is entirely dependent on the user's input. Still, this system has some nice features such as sending a text to the desired employee.

"Chui - The World's Most Intelligent Doorbell" is one of all the web-based doorbells for the private home. Most of them comes equipped with a camera, microphone speaker, and wifi for Internet connection. This allows the owner to throw an eye on the front lawn at any time and answer the door even if the owner is not actually present. There are however some products that have taken this to the next level. Chui is a soon to be launched product that promises a great deal. This is their product description:

Chui is an intelligent doorbell that uses facial recognition to make your home keyless, secure, and individualized. With Chui, your face becomes a key that can unlock the front door, while enabling connected devices tailored to your preferences. Relax in confidence, knowing Chui will automatically notify you and your loved ones of who's at the

door. Sleek and stylish, Chui does more than enable a house that is smart – Chui shapes a home that is you.

(Chui 2015)

No matter how great this product is claimed to be, it is still designed and developed to serve a private home and not an office; hence it does not provide the necessary functions to work as a receptionist. The user, or in this case the visitor, has actually no control of the interaction. As soon as the face is recognized or the button is pushed the user's ability to influence the product is gone and the responsibility is all up to the product and the owner.

There are of course other similar products in both the private and business sector with similar functions as the ones described above. E.g. Ring, a smart doorbell that uses motion detection. Meaning it can double up as a security camera for the front lawn if desired. As has been discussed above, there are several existing products but many are either outdated or lack customization options, both for the owner and the visitor. Furthermore, only Chui provide a somewhat personalized user experience by making use of facial recognition.

3. Theory

What makes a virtual receptionist? To make it work, different fields need to come together such as identification, presentation, interaction between user and receptionist, and context awareness - both the spatial and the personal aspects; all to create a credible, enjoyable, and beneficial product.

3.1. Identification

To achieve a high-fidelity prototype in the relatively short time, this project apprehends existing frameworks in most, if not all the aspects. Working this way allows us to quickly build and prototype new designs and features. To make the project comprehensible, each piece of technology will be implemented as an individual module, making it flexible and scalable in any direction. The initial design is basically divided up into three separate domains. Analyze the visitors, distributing data between the modules and illustrating data.

A vital feature of the project is to detect the people that move in and out of the entry hall and detect who they are. This can be done with a variety of technologies and techniques, each one with their pros and cons. Fundamentally there are two different approaches to identification. Either one can sense biological attributes of the visitor, or by sniffing digital identifiers that the visitor carries. In this section some of these technologies are evaluated.

3.1.1. Facial recognition

Facial recognition is divided into two steps; detection and identification. The biggest issue with facial detection and recognition is that faces vary greatly in shape, form and expression. Many different methods have been developed to detect a face within an image; using everything from contours, skin tones, template matching, etc (Wilson and Fernandez 2006). However, the most popular and efficient way of detecting faces today is by using Haar-like features. The technique extracts contrast patterns from the image into identifiable shapes. To create a face, a couple of these feature areas need to be placed in a certain

structure but not necessarily look exactly the same way. The relative placement between the eyes, eyebrows, nose and mouth is more important than their exact shape and form (Wilson and Fernandez 2006).

The next step is identifying, to whom the face belongs to. This is usually done by comparing the new face with the set of already known faces. The accuracy of the recognition is influenced by many factors, such as choice of matching algorithm, image quality, image similarity, lightning, etc. In terms of accuracy there was a big leap recently (2014) when the system DeepFace achieved an accuracy of 97.35 % (Taigman et al. 2014) when running the LFW database (Labeled Faces in the Wild) which consists of 13 323 web photos of 5749 celebrities (Huang et al. 2007). This can be compared to an average human facial recognition, shown the same cropped image humans had an average result of 97.53 %, virtually the same. However, this comparison is not entirely fair as humans rarely ever have to identify a person in a cropped image. When the study participants saw the whole picture, the accuracy increased to 99.20 % (Kumar et al. 2009). Unfortunately, the DeepFace api is not available on the market.

There is an open source computer vision library called OpenCV which has it's own implementation of facial recognition called FaceRecognizer. It uses the algorithm Eigenfaces and Fisherfaces to identify people. Eigenfaces recognize a new face by utilizing a data set of faces and describes it by comparing resemblance with the data set. Each face is represented by a multi-dimensional vector where each axis maps to a given feature - as in "your face is 17 % similar to eigenface A, 63 % to eigenface B, and 20 % to eigenface C". Based on that it decides on who it is (opencv-dev-team 2014). FisherFaces complement eigenfaces, as eigenfaces are comparing contrasts between images it means that eigenfaces is sensitive to lighting differences in images. The Fisherface algorithm analyses difference in contrast but is not as sensitive to light making it more reliable when there is non-perfect image conditions. (Belhumeur, Hespanha, and Kriegman 1997) It is difficult to foresee what kind of results one could expect from an implementation. It is most likely dependent on the tweaking of the algorithms and what the lighting situation will be.

In our situation, we will have a small room with no windows and good lighting from the ceiling. Each person will enter the room by either walking out of the elevator, most likely facing forward or coming out of from the office to go to the elevator. This makes the situation ideal as we can foresee how people will most probably move in the room. For instance, having one camera facing the elevator and one facing the door would make it easy to detect that someone is coming and from where, adapting the experience to each situation.

It would be possible for us to create or modify existing OpenCV applications for face recognition. However, our initial experimentation indicates that the iden-

tification can be uncertain and does not scale easily. The alternative that was utilized was to make use of one of the numerous web-services offering face recognition.

Face++ is one of these web-services that offer facial recognition. The company behind the service, Megvii just released a new article, where they claimed to have achieved 99.5 % accuracy on the LFW-dataset with their algorithm, which is a better than humans (99.20 % at its best as mentioned previously) and a new benchmark record. However, it is worth to mention that they tried another data set CHID (Chinese ID), where the performance were below acceptable, which shows that the technology can still be unstable.

3.1.2. Mobile sniffing

Another method of identification is taking advantage of the fact that everyone is carrying a cellphone. By capturing the cellphone's unique id and connecting it with a name it's possible to track a person. However, this requires one to jam anything but GSM signals since more advanced protocols use encryption. Due to lack of time and the questionable legality, we didn't pursue this method of identification, but it could have had great benefits compared to facial recognition such as instant identification as compared to facial recognition where we need to send an image across the world, process it and wait for a response. The signal is not as sensitive to conditions while the camera needs, to be in focus, the face needs to be visible, and face towards the camera.

3.1.3. BLE

Bluetooth low energy, or Bluetooth Smart as it is also called, is developed by the Bluetooth special interest group. BLE is similar compared to classic Bluetooth, in range and data capacity but is designed to, as the name implies, be more energy efficient. The purpose is to be able to keep connection for extensive periods of time without draining the battery. This also means that BLE is always active and can detect within a distance of approximately 100 meters. The only downside to this is that only the latest cellphone models have BLE, it is however predicted that by 2018, 90% of all devices with Bluetooth will have BLE support (SIG 2015).

3.1.4. Wifi-sniffing

By implementing a sniffing application it would be possible to detect activity on Interaktionsbyråns network, record the mac-address of the device and map it to an individual. This method of identification is not likely to be as quick as BLE but it could work in parallel with other identification services to keep track of if and where a person is in the office. A major downside to wifi-sniffing is that it requires that the person has been at Interaktionsbyrå before and used the WiFi.

3.2. Ambient Intelligence and Context-Aware Computing

Ever since Weiser (1991) presented his vision of ubiquitous computing, researchers have worked toward the goal of merging the physical and the digital into one seamless experience. In pursuit of this goal, scientists have studied a variety of approaches. Ishii and Ullmer (1997), under the label of tangible interaction, wanted to give physical form to digital information. In contrast virtual reality¹, pioneered among others by Sutherland (1965), tries to envelop the user completely in an artificial environment, making it feel as though the user has been transported into the digital domain.

As a compromise between the two approaches above, augmented reality (Manovich 2006) overlays physical space with dynamically changing information. While it can be made to envelop the user, more commonly the digital aspects are presented on screens, reminiscent of Alberti's window into a different world (Alberti 1435). By extending the physical world with digital information a richer, multi-dimensional, environment is created. One example of such an environment is the shopping mall Langham Place, Hong Kong. In the roof of the Grand Atrium the 'Digital Sky' (ARUP 2015), a huge screen constructed using more than 200 projectors (Wang 2004), displays ever changing visuals for the consumers below.

In his article "The poetics of augmented space" Manovich (2006) frames the use and construction of augmented space as an artistic and architectural problem. This perspective builds upon the idea of "architecture as communication for the Information Age (rather than as space for the Industrial Age)" formulated by Venturi, Brown, and Izenour (1972). Communication, however, carries little intrinsic value except the subject which is communicated. While Venturi arguably saw the architectural communication as mainly between architect/artist

¹Virtual reality can actually trace its roots back to as early as the mid 19th century and panoramic murals such as Sala delle Prospettive (Grau 2004).

and viewer, augmented spaces facilitates communication between computer and viewer. As such, an important question becomes how one can imbue a computer with the ability to judge what information is relevant to the user at a given moment?

Ambient Intelligence (Aml) refers to “electronic environments that are sensitive and responsive to the presence of people” (Aarts 2006). The IST Advisory Group (ISTAG) expands on that definition by describing Aml as an environment where:

People are surrounded by intelligent intuitive interfaces that are embedded in all kinds of objects and an environment that is capable of recognizing and responding to the presence of different individuals in a seamless, unobtrusive and often invisible way.

(Ducatel et al. 2001)

One can view Aml as a special case of augmented reality where the digital element is aware of the context of use and actively tries to anticipate and adapt to the individual. It's realization depends on the development of two additional technologies. Firstly, data need to be collected about the context and stored in a way which allows the computer to interpret it. A good definition of “context” is that of Abowd et al. (1999):

Any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves.

(Abowd et al. 1999)

In the words of (Bazire and Brézillon 2005) the context imposes “a set of constraints that influence the behavior of a system” which hints toward the second dependency of Aml - artificial intelligence. If the contextual data embody the semantics of an ambient intelligence system, then artificial intelligence defines and interprets the politics of the same.

3.3. Natural interaction

Machines that fit the human environment instead of forcing humans to enter theirs will make using a computer as refreshing as taking a walk in the woods.

(Weiser 1991)

Continuing on Weiser's insightful vision. He described a world where computers have been completely integrated into our lives. As the computers sense and adapt to our needs and behavior their use is made more enjoyable and allows us to be more productive. His vision has guided much of computer research ever since. Then how come IT-products are still far from natural, in the way we interact with them and their ability to make smart decisions based on our needs, 24 years later?

A short answer would be that it is more complicated than originally thought, and there are many reasons for this. Ishii and Ullmer (1997) made the distinction between bits and atoms as being the basic constituents of two fundamentally different worlds. What we want to achieve is to both bridge bits from the digital world making them as tangible as atoms in the physical world and to make atoms as adaptable and intelligent as digital bits.

GUIs fall short of embracing the richness of human senses and skills people have developed through a lifetime of interaction with the physical world.

(Ishii and Ullmer 1997)

The user's sense of naturalness is not only dependent on the manner of interaction but on the whole apparition of the design and presence of the artifact. The experience is then defined by it's level of physical inconspicuity contra its level of experienced potential. *"Of course, tools are not invisible in themselves, but as part of a context of use"* (Weiser 1994). In this sense the goal is not to make the constituting design components invisible, but making sure that each part is focused on and contributes to the overall aim of the product. When natural interaction is discussed it seems that the relationship between tool and user is often mistaken for a fixed state. However, the majority of pure physical tools requires a significant amount of practice to master. Ergo the tool becomes an extension of the user and not a barrier to her own potential. It suggests then that we have not only created a world adapted to us, but have ourselves adapted to the world in the strive of fulfilling our potential.

Continuing on the topic, the naturalness of any interaction with objects can be challenged. One could argue that until we have an AI that could interact on the social level of humans, that is - pass the Turing test (Turing 1950), can we really claim to have achieved a truly natural user experience?

3.4. Physical Space

Place is security, space is freedom: we are attached to the one and long for the other.

(Tuan 1977)

To understand how we can create an interesting and exiting encounter, we must first understand the space and the context we are going in inhabit. To see how the device affects the space and how the space affects the experience. However, Tuan (1977), Harrison and Dourish (1996) argues that the actual space is irrelevant and what we actually strive to understand is the concept of the place and its correlation with context.

Appropriate behavioral framing, is not rooted in the properties of space at all. Instead, it is rooted in sets of mutually-held, and mutually available, cultural understandings about behavior and action. In contrast to “space”, we call this a sense of “place”.

(Harrison and Dourish 1996)

In our case, the space within which our design is going to live is a small bland hallway. For most people this might not be the most interesting room at the studio but it is the first that a visitor or employee encounter, which means this room welcomes people, sets the atmosphere and mindset, and could give the first impression of the company. Still, there are a few predefined rules that is expected of a hallway. In order for the room to make sense we can not only look for the potential of the space, but must also consider it as a place with existing expectations in order to design a powerful and instantaneous experience.

One eloquent reflection on the difference between space and place is done by physicist Niels Bohr in conversation to Werner Heisenberg at the Kronberg Castle in Denmark:

Isn't it strange how this castle changes as soon as one imagines that Hamlet lived here? As scientists we believe that a castle consists only of stones, and admire the way the architect put them together. The stones, the green roof with its patina, the wood carvings in the church, constitute the whole castle. None of this should be changed by the fact that Hamlet lived here, and yet it is changed completely. Suddenly the walls and the ramparts speak a quite different language. The courtyard becomes an entire world, a dark corner reminds us of the darkness in the human soul, we hear Hamlet's "To be or not to be." Yet all we really know about Hamlet is that his name appears

in a thirteenth-century chronicle. No one can prove that he really lived, let alone that he lived here. But everyone knows the questions Shakespeare had him ask, the human depth he was made to reveal, and so he, too, had to be found a place on earth, here in Kronberg. And once we know that, Kronberg becomes quite a different castle for us.

(Niels Bohr, as recounted in Mills 1976)

The place is then not only influenced by the context of its relative position in the building, giving it its purpose; but also a response to the expectations a visitor has, such as what happens when the elevator door opens, personal opinions of the company, what kind of mood the visitor is in, etc. The place is an individual creation, through the design we make, which replies to the expectation in various ways. This insight highlights the value in considering potential interpretations. All this implies that we have to consider the experience of the user when interacting with Ameer relative to the full context of the room where the interaction is carried out.

4. Methodology

To reach our final prototype, strategical choices have been made. Choices such as the overall development plan, how to handle feedback from stakeholders, what materials should be used, etc. These choices has become the backbone of the prototype and gives a idea of how the prototype came to be.

4.1. Theoretical frameworks and methods

The project is carried out according to a number of theoretical frameworks and methods. These ideas constitute the theoretical underpinnings through which our results are achieved and interpreted.

4.1.1. Action Research

The project is carried out in the manner of action research (Berg, Lune, and Lune 2004). Through short iterations of exploratory action and analytical reflection a double loop learning process (Argyris 1991) is achieved, inspiring and informing later activities. Put a different way, in working on each design task we quickly alternate between ideation and informal testing of rough prototypes. However, the ideas tend to be more numerous and radical toward the start of the process. Conversely, toward the end, the prototypes become more polished and new ideas tend to stay closer to the existing concept. Laseau (1980) illustrated this process of ideation and critical analysis as two cones, opposite and superimposed on each other (see figure 4.1). While this model is not something we explicitly attempt to follow, it is a fairly accurate description of our process both over the project as a whole and within individual iterations.

Given the exploratory nature of the project it should be viewed as a cursory investigation into an unfamiliar design territory. Through working with the specific case of designing a digital receptionist for Interaktionsbyråns offices, we will try to tease out interesting avenues for further research and gain a good sense of what challenges are coupled with the more generic task of designing a context-aware digital service agent. The important result of the project has

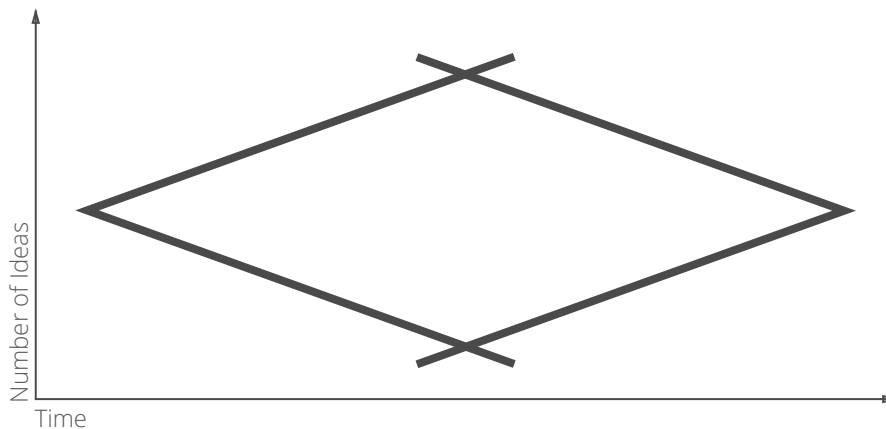


Figure 4.1.: An interpretation of Paul Laseaus design process model; a good visualization of our creative process on both a macro and micro level.

more to do with the idea development than the finished product. As such, this project is an example of research by design where we tap into the designerly ways of knowing first defined by Archer (1979) and later developed by Cross (2006).

4.1.2. On-site development

As the work is carried out at and, to a certain extent, together with Interaktionsbyrån the design will be adapted to their specific needs. In this way the resulting design is not simply a product of hypothetical scenarios but a synthesis of *a priori* reasoning and empirical testing in a real-world setting.

Developing the project at Interaktionsbyrån directly and being welcomed as a part of their organization allowed us to get a deep understanding of what they expected of the project. We choose to upload all our progress on the screens in the lobby directly. With the philosophy *something is better than nothing*, meaning that it is better to show a poorly performing software or feature than showing nothing at all. This gave the 20-30 designers and developers working at the studio some insight of the progress and potential of the project. This worked well as communicative tool to get in contact with the employees, their thoughts and ideas.

4.1.3. Microservices

In order to allow the system to grow and evolve, it's vital that it has an architecture that is fault tolerant and flexible. This will be achieved by designing a system that is divided into a number of microservices (Birman and Joseph 1987), where each module can be scaled or exchanged individually with minimal effect on all the other nodes. The modules will communicate in an event driven manner over a shared message bus implemented with the MQTT transport protocol. This means that new nodes can be added or removed depending on the situation.

4.1.4. Rapid Prototyping

There was an ambition to implement all that was designed. Therefore, we needed a quick way to try tools, technologies, features and designs. Either by exploring different designs by sketching or make proof of concept regarding a tool or a feature. There were two questions, and they were asked in the order: *is it possible?* and *is it good?* These questions forced us to make a big quantity of prototypes as each prototype needed to pass the question. Rapid Prototyping is done through many iteration with a high tempo. Each prototype is evaluated and then refined; with new knowledge and ideas the prototype is improved or a new prototype is made (Cerejo 2015). This method have been adapted freely for most prototypes, most of the prototypes are microservices with a specific task to do it. If it could perform the task according to our benchmark the microservices was deemed satisfactory. However, the services that are in contact with the graphical and user experience went through textbook standard rapid prototyping. It took many iteration and a couple of complete turnarounds before settling.

4.1.5. Continuous Delivery

Continuous delivery (Humble, Read, and North 2006) is a term for having implemented a very fast pipeline between development and production. Basically it means that you provide the user or client with a new version of the software every day or even more often. This creates transparency in the project but also requires a comprehensive suite of automated tests to verify release integrity. In this project, new features have been pushed to the production server several times a day and it has proved to be a worthwhile strategy. Having the latest version of Amee running in the lobby helps facilitate discussion and makes sure everyone knows about the state of the project.

4.1.6. Automated testing

Automated testing has a number of advantages in software development (Huizinga and Kolawa 2007). Firstly, having a suite of automated tests means it's now trivial to verify the integrity of a large number of situations all at once. Furthermore, it provides a sense of confidence and security to the programmer as he or she can easily make sure that any new features do not introduce new bugs or break existing functionality. This is especially helpful if a programmer is not fully versed in the project yet. Furthermore, the tests allow for Continuous delivery (section 4.1.5). As many benefits as automatic testing have, though, it also introduces a not insignificant overhead to development.

4.1.7. Test-driven development

Traditionally, software has been written first, and afterwards tests are written to verify it. Test-driven development shifts these activities so that first the tests for a given api is implemented, and then you develop the software to meet this specification (Ashbacher 2003). This change forces the developer to think about the integration early on in development and provides an objective quantitative measure of feature completion.

4.1.8. 100 ideas in an hour

A version of IDEO's brainstorming guidelines, brainstorming is a good method for initial ideation, where quantity is the most important outcome. Encouragement, never judging. For this, IDEO has put up seven guidelines or tips as they call them. In our case we had sessions for 15 min and then 5 min breaks (OpenIDEO 2011).

4.1.9. User stories

The user stories (Cohn 2004) was made as a sounding board for the design of the prototype. An easy method to make sure that our design achieves our hypothetical theories of what different stakeholders would hope to get out of Ameer. The stories was based on on the previous feature ideation and summed up to 29 stories (p. 56) and one scenario (p. 64).

4.1.10. Interviews

In an effort to evaluate the final state of the prototype, five interviews were carried out, both with employees and first-time users. This exercise aims to get a qualitative insights into the users' experience of Amee. The findings can then be used to guide later design and development.

4.2. Engineering Tools

In working with the project we used many productivity-enhancing tools and frameworks in order to facilitate rapid prototype development. Furthermore, the tools abstract away some of the complexity in the engineering task allowing us to work at a higher conceptual level. Our choice of tooling has definitely had a big influence on what has been possible to do and how we work. Conversely, how we want to work has influenced what tools we've chosen - blurring the line between method and tooling. Below are listed some of the most important tools used in the project:

4.2.1. Node.js

As stated on their official homepage:

Node.js is a platform built on Chrome's JavaScript runtime for easily building fast, scalable network applications. Node.js uses an event-driven, non-blocking I/O model that makes it lightweight and efficient, perfect for data-intensive real-time applications that run across distributed devices.

(Joyent 2015)

It will form the base for most if not all other development and is the key technology behind most other tools and frameworks in this section.

4.2.2. Brunch

Brunch is a modern build tool for the web running on top of Node.js. It can help compile, lint, minify, generate source-maps and optimize files for production. Through Node.js'IO-streams it keeps build times comparably low and a

‘convention over configuration’ approach to the build pipe allows for terse configuration files (Brunch 2015).

4.2.3. Mosca

The MQTT protocol is a simple to use transport protocol which uses a publish/subscribe communication model and transparently manages unreliable connections (Banks and Gupta 2014). Mosca (Collina 2015) is a Node.js implementation of the central message broker which handles forwarding messages between subscribers. Being flexible, simple to use, very fast, and provides websocket support out of the box, it’s an excellent MQTT broker.

4.2.4. AngularJS

As a lightweight open source project initiated by Google, AngularJS has grown to be a leading templating engine in modern web applications. It’s main features, being bidirectional data binding where the model data is automatically kept in sync with the view and being a fat client which means it process data locally which releases pressure from the server.

4.2.5. Node-RED

Node-RED is developed by IBM Emerging Technology division and is a semi graphical development tool to connect hardware and software together. It is semi graphical in the sense that the whole application is built on a GUI where all logic is split up into small separate graphical nodes. Where each node is either preprogrammed or have to be written from scratch. Node-RED will be the development tool to listen and write to Slack as there is already basic support.(IBM Emerging Technology 2015)

4.2.6. OpenCV

OpenCV is a open-source library for high efficient realtime applications. It is well established in image processing and analysis including some tracing algorithms. In this thesis Haar-cascade¹ has been utilized to trace face detection

¹Haar-cascade refers to Haar-like features which are features found in digital images when pixels are represented as a numbers or as a signal. The features are then areas of contrast where numbers differ more or less towards one another.

in a live video stream resulting in system awareness of human presence (Itseez 2015).

4.2.7. Qt

The initial plan was to develop the GUI as a web application. This changed however, when we together with Interaktionsbyrå decided that it would have a greater effect on users if there were three screens instead of one. It would also allow us to convey with higher fidelity the sensation of a digital wall/living wall not just a screen with an interface. However, we were unsure how well the performance would be if the application would be developed in a web-application. Instead Interaktionsbyrå recommended Qt, a cross-platform development application that uses C++, QML, and JavaScript. The first impression indicated that it is easy to get going and learn (Company 2015).

4.2.8. Hardware

The final Hardware we ended up using is three 46" screens from Liyama where one of them is touch. These are driven with a small computer that is placed in the ceiling. In total the screens span more than 3.2m from side to side. The computer runs all the microservices e.g. amee-greeter, amee-nodered, amee-cam, etc. (All the microservices can be found in Result 6 on page 33).

4.2.9. Git

The now close to ubiquitous code versioning system Git makes collaboration between developers much simpler (Git 2015). Written by the father of the Linux operating system Linus Torvalds, it records changes on individual lines rather than whole files making it much easier to merge changes from different sources. In our project everything from microservice code to the latex thesis sources are stored safely and versioned with Git.

4.2.10. Mocha.js / Chai

To simplify test development and running of automated test (section 4.1.6) we use the Mocha.js framework (Holowaychuk 2011) in combination with the Chai assertion library (Chai 2015). Chai provides a readable syntax for specifying test conditions that must hold for a test to be considered successful.

4.2.11. Preprocessors

A common trend today is to use precompilers to provide syntactic sugar to existing languages. This increases development enjoyment and productivity by increasing the expressiveness, readability and automating the inclusion of non-essential syntactic details. For these reasons, we use precompilers throughout most of the project to write in Coffeescript (CoffeeScript 2015), Stylus (Stylus 2015) and Jade (Jade 2015) instead of the usual Javascript, CSS and HTML.

4.2.12. Ubuntu

Ubuntu is the most popular open source operating system today (Ltd. 2015). It's based on Debian Linux and in this project it's used to power the server on which most of our microservices run. It provides a secure, easy to use, developer-friendly environment to host our application.

4.3. Planning

In preparation of the project, a gantt chart detailing the expected project progression and activities was developed. It can be found as figure 4.2 on page 26. Just like the project as a whole can be divided into the three overarching phases "planning", "execution" and "writing" each iteration follows a similar structure.

At the start of the project, it was assumed that the receptionist application would probably consist of a large wall-mounted touch screen mounted in the lobby, though we were aware that this was likely to change in design and features as the project progressed. Due to this expectation it was decided that a flexible architecture was needed to support changes throughout the work. Furthermore, it was deemed likely that use cases could shift over time and that it might be relevant for some components to be used in multiple contexts.

As can be seen in the Gantt chart, the first and primary inlet to be implemented was a face recognition service which can detect and identify people entering the lobby area. Other possible inlets included a Wi-Fi MAC-address collector and a social network data scraper. Regarding outlets, in addition to the receptionist application, an administrative website for use within the company to manage the collected data was to be developed.

Figure 4.2.: The planned activities and durations over the course of the project.

5. Process

The development of the prototype is logically divided into two different aspects, the technical platform that dictates what features we are able to provide, and the design which includes GUI and UX design. Together they make a whole that should be compatible with not just each other, but also the general goal of Amee.

To realize this goal, an initial plan of the architecture was created to get the project going. The plan involved a general framework centered around a database of circumstantial information. The database should be connected to a number of information inlets, that aggregates data, and outlets, that makes use of the data e.g. the receptionist screen. This microservice architecture (see section 4.1.3 on page 20) was imagined as something similar to figure 5.1 and 5.2.

Even though there was a plan, it was quickly realized that the technical aspect and the design aspect was challenging to balance, as they are both dependent on each other while having fairly separate development processes. The benefit of developing them in parallel has been flexibility. By quickly adapting both aspects to one another when realizing new potential or limitations of one aspect. However, developing this way creates a greater risk for scope creeps as features need to be implemented or demoed before they can be added to the feature list. And if they are not on the feature list it is inappropriate and difficult to decide on a design. This means the GUI had to be designed general enough that features could be added and removed easily until the final implementation. It should also be mentioned that the whole development of the project has had a natural change of focus as our understanding of the problem have evolved.

There is a common understanding of what a physical reception is and what it can be used for. This understanding is expressed in the design, the context and the spatial placement of the reception. However there is no common agreement on how a digital reception should look and behave. This means that there is no placement or design guidelines that creates instant recognition on what The-GreatWall is, which means that the design of the product should be expressive enough to either make anyone passing by interested enough to further explore it or make it self-explanatory by the first glance.

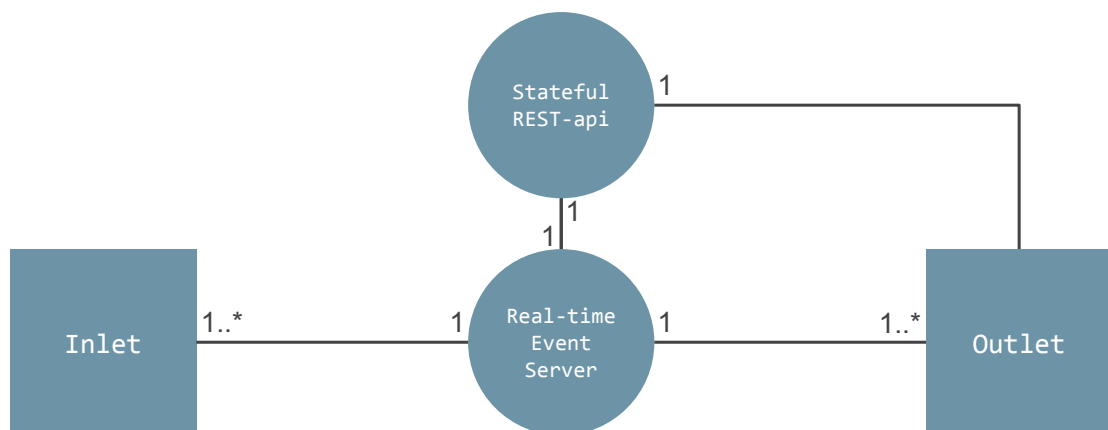


Figure 5.1.: The conceptual model of the receptionist system.

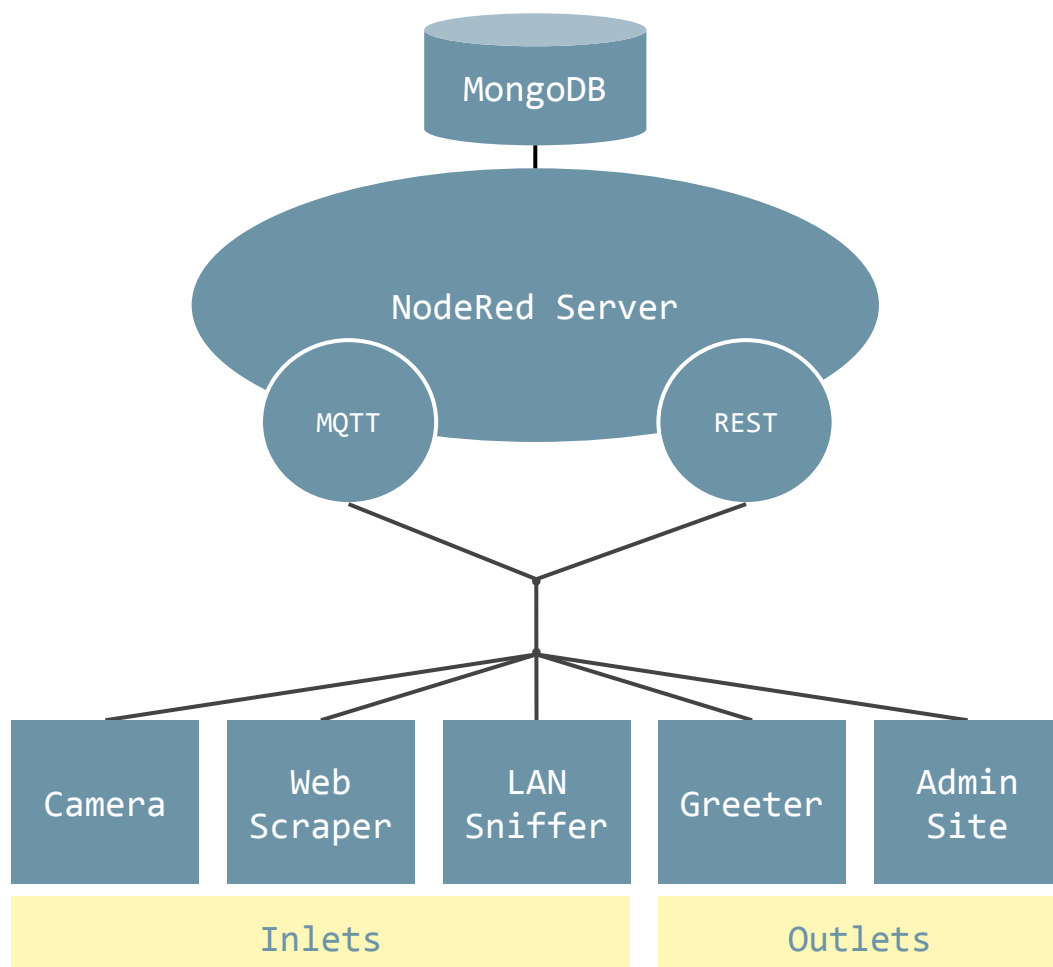


Figure 5.2.: A component diagram of the receptionist system server.

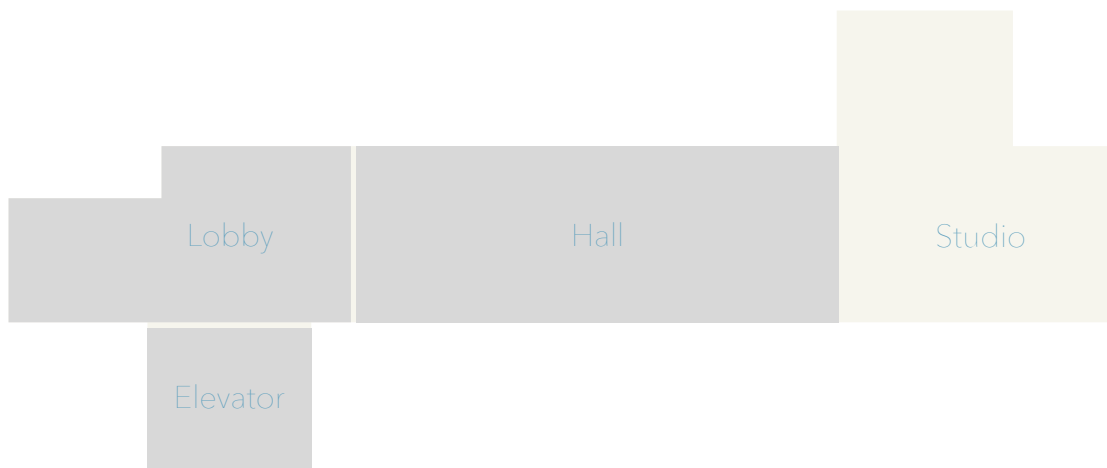


Figure 5.3.: The layout of the relevant spaces as seen from above.



Figure 5.4.: Traditional GUI sketch

The first plan was to place a big touch screen outside the entrance facing arriving people as they come out of the elevator. The initial strategy was to treat it as a traditional large touch interface built using common metaphors such as windows, buttons etc.

However, we did not want Amee to be perceived as a traditional application which lead us to looking at alternative interface metaphors. We investigated turning the whole wall into an interactive screen using multiple monitors arranged into a grid. We ended up buying three large monitors, one in the lobby and two in the hall, placed close to each other in order to create the sensation of one very wide screen covering the wall. The lobby and the hall are separated by a glass divider, making it possible to see all screens from any position. However, the other side's accessibility is limited due to the physical glass barrier, distance and reflections in the glass.

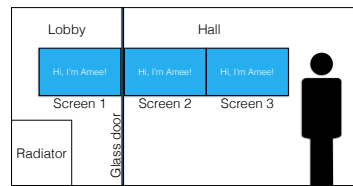


Figure 5.5.: Images illustrating on how the screens will be perceived in the lobby and hallway

As there is a glass door disrupting the “seamless” screen there is a functional difference between the screens in the hall and the screen out in the Lobby. Depending on the user’s location, the screen in the same room will serve to give practical service while the screen/screens on the other side of the glass door is more fitting to illustrate abstract concepts such as room dynamics, when the door is opened contra when the door is closed or information flow in the studio.

This generated a new philosophy. Instead of taking the classical approach, to stack the information in an as space efficient manner as possible, we developed a user centered concept. A normal GUI is limited by the dimensions of the monitor and strives for space efficiency by making good use of its borders to contain all the information. With a conceptually unlimited canvas, designing for borders and containment does not make much sense. Instead the limitations are reachability and focus area. The goal is to make an impression on the user while not overwhelming him or her with information.

A normal interface is technically designed according to a top-left reference point. In our concept, the main reference point consists of the user’s position. Therefor, we imagined a GUI which would grow organically outwards from that point. The most important and general content could be placed centrally while less important content would be located in the peripheral. This means the user should not have to adapt his or her position to get access to the personal service and content. Instead the interface moves with the user and is ready to serve when and where the user wants to.

While working with the infinite canvas and organic window layout, we still felt like the prototype lacked a certain human quality. Our solution was to make the

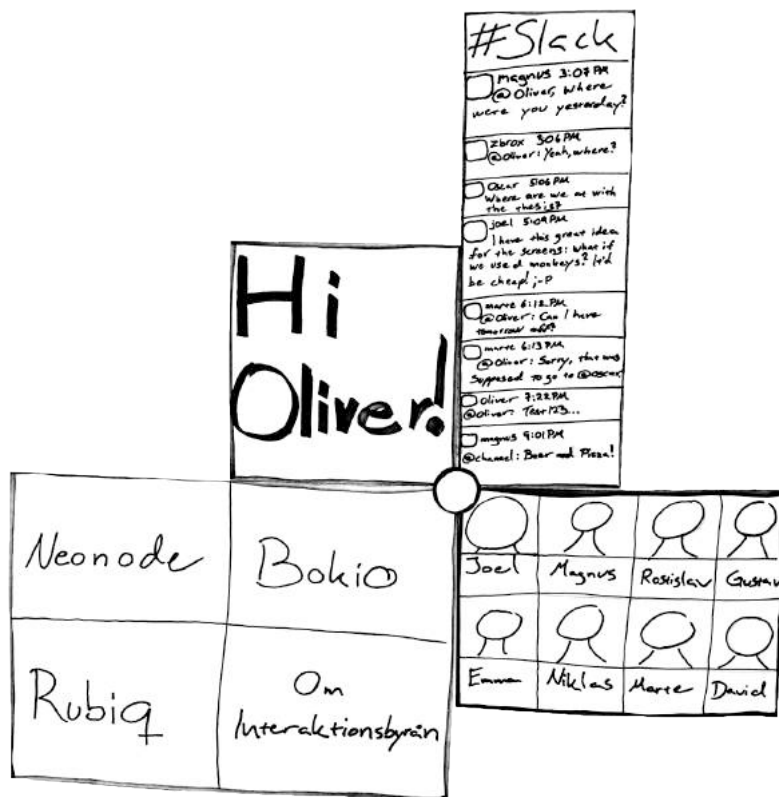


Figure 5.6.: Personal GUI Interface



Figure 5.7.: Personal GUI interface animation



Figure 5.8.: Conversational GUI interface animation



Figure 5.9.: Early alpha of prototype and all hardware mounted

interaction conversation based - similar to talking with a real receptionist. This would also allow the interface to be more of an active participant in the interaction. It does not simply wait for user commands but instead takes initiative when appropriate.

This lead us to the conclusion of a wall that spans three screens, with an interface adapted to the user, where the interface is the active participant in a conversation. This lead us into a sequential time based interface. Where the conversation is divided up into sequences. The ambition is to give the impression of a fluent conversation. Wether the user interacts or not, TheGreatWall will never stop to require an interaction from the user. However if the user chooses to, he or she can take control of the conversation.

6. Result

The results of the project are twofold. There is the technical platform on which the virtual receptionist is built, and then there's the interactive prototype of the receptionist itself. In this chapter we present the makeup of each of them.

6.1. Platform

Powering the interactive virtual receptionist is a larger ecosystem of microservices (Lewis and Fowler 2014), each with their own limited concern and communicating over MQTT. At the end of the project, we have a total of 15 subprojects making up 11 distinct services that run concurrently. Most services are implemented as Node.js (Joyent 2015) scripts unless otherwise stated. Below are short descriptions of each implemented service and figure 6.2 shows a visual overview of the system.

6.1.1. amee-mqtt

A small service which configures and launches a Mosca (Collina 2015) MQTT-server backed by a Redis database. The server is configured to accept both standard MQTT messages and websocket connections. This allows us to establish connections from client side javascript like in a website as well as normal MQTT clients.

6.1.2. amee-nodered

This service hosts a standalone Node-RED (IBM Emerging Technology 2015) server. It's primarily responsible for integrating Slack (Slack Technologies, Inc. 2015) - a popular team messaging application used at the Interaktionsbyrå studio - with the MQTT communication bus. The integration allows notifications to appear on Slack (as if written by the virtual receptionist) and messages and commands can be issued directly from Slack instead of a separate administration interface.

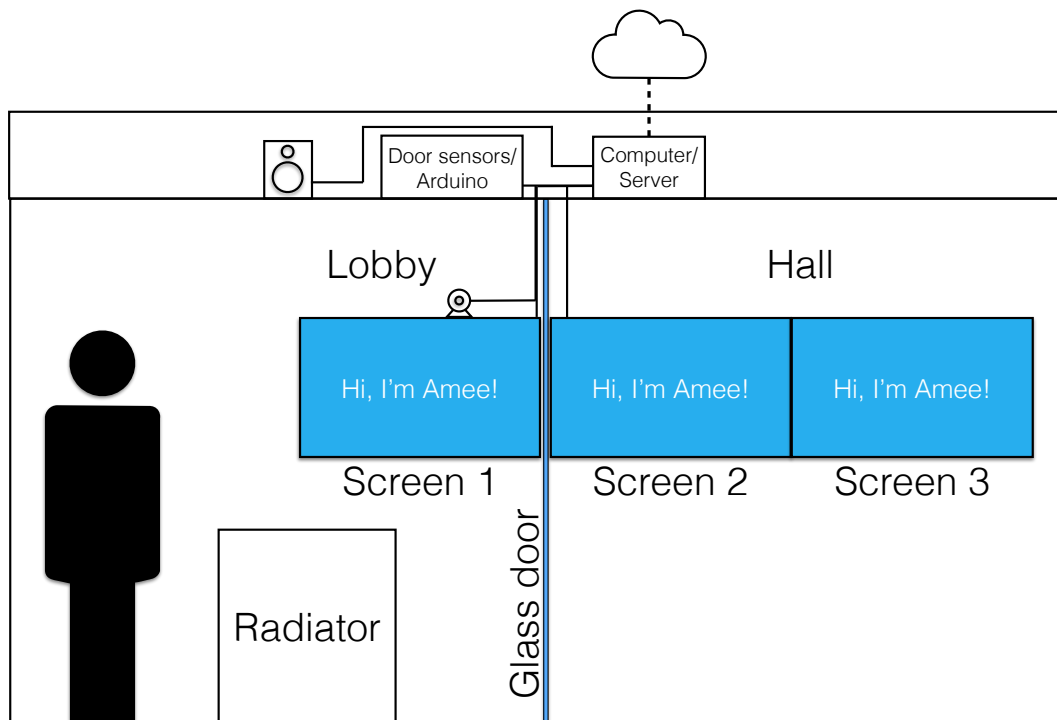


Figure 6.1.: A wall elevation of the entrance

6.1.3. amee-cam

A webcam is set up in the lobby and this microservice processes its images in realtime using the open-source image recognition library OpenCV (Itseez 2015). When it detects a face in the picture, it stores a picture of the person and notifies other services over MQTT. The service is implemented using Python (Python Software Foundation 2015).

6.1.4. amee-doors

Connected to the elevator and lobby doors are small sensors that can detect whether the door is open or closed. The sensors are read by an Arduino (Arduino 2015) which in turn acts as a slave for a Node.js process using the Johnny-five (Bocoup 2015) library.

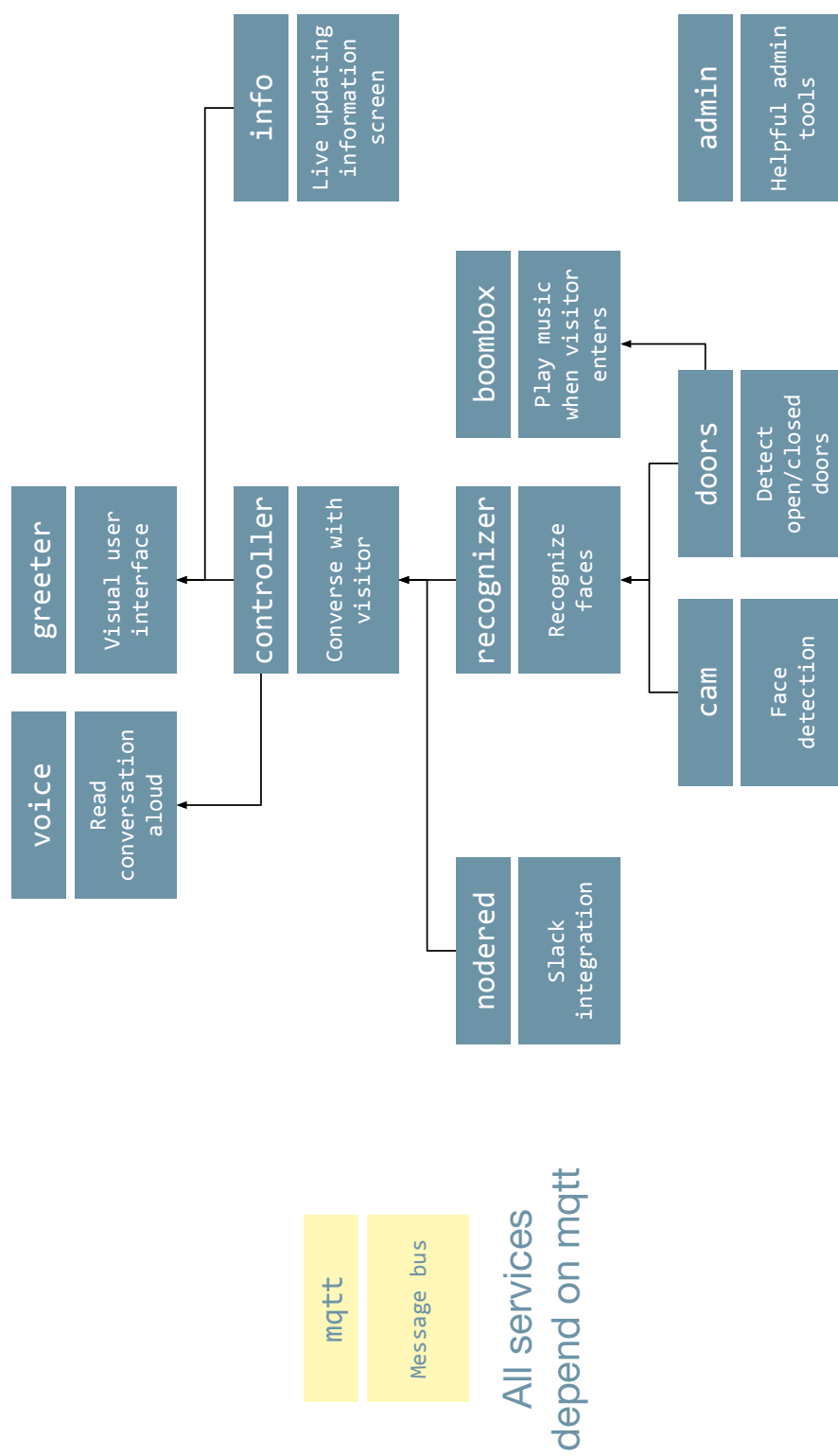


Figure 6.2.: A dependency model of the microservices that make up the technical platform. Arrows denote a dependency relation with the pointed end facing the dependent service.

6.1.5. amee-admin

The admin service runs a simple express (Express 2015) server serving a cursory implementation of an administration website implemented with modern web technologies such as AngularJS (Google 2015a) and Brunch (Brunch 2015). At the end of the project, the website work as a live-updating MQTT message monitor for the communication bus.

6.1.6. amee-recognizer

This microservice listens for detection events emitted by the amee-cam service, described in section 6.1.3. When a detection happens, the service transmits the picture taken to the face++ (Megvii, Inc. 2015) web service for further processing. Face++ tries to identify the person in the photo and will eventually respond with a list of potential candidates. The recognition service then publishes those candidates over MQTT for other services benefit.

6.1.7. amee-boombox

The boombox service plays a random piece of music when a person enters the lobby from outside. It listens to events from the door service (section 6.1.4) to determine whether people are entering or exiting. Each song is played for about 30 seconds and the volume is gradually lowered according to a quadratic easing function.

6.1.8. amee-greeter

This service implements a dumb user interface managed by the amee-controller service (described in section 6.1.9) over MQTT. The interface is displayed over the 3 side-by-side monitors mounted at the entrance to the studio. It implements a number of views for displaying various kinds of data e.g. simple text, images, questions and websites. The service is implemented using QML and Qt (Company 2015).

6.1.9. amee-controller

The controller service controls the greeter view through MQTT messages and implements a basic conversational engine based around a custom state-machine library. It uses an asynchronous event system to collect input and other information and then transfer it synchronously to the current state in the state machine. Each state individually process the data and decides if and how to respond to a given event. In this way we create a basic context aware AI that can respond to different kinds of external stimuli according to a number of pre-defined cases.

6.1.10. amee-voice

To make the virtual receptionist feel more “alive” we created a service which unobtrusively listens to the messages sent by the controller service and reads them out aloud using a text-to-speech web service provided by Google (2015b).

6.1.11. amee-info

Most of the implemented user interaction in the prototype happens in the lobby outside the studio door. In order to bring life to the two inner screens, this service was set up to poll Eventbrite (Eventbrite 2015) for information about coming events at Interaktionsbyrån and display it on the hall monitors. The information is compiled into a single page website, sent to the greeter and loaded into a webview. This allows considerable freedom in changing the displayed information and design without modifying the greeter code.

6.1.12. Custom libraries

Besides the above microservices a number of libraries and tools were implemented in order to further improve the modularity and reusability of subcomponents of the project.

amee-scrapers

This library and command line tool is used to collect some basic information and profile pictures from common social networks. All of the images and information gathered are publicly available to anyone with an internet connection and a browser. For the prototype, only scraping of a single social network was fully implemented.

amee-faceplusplus

The image processing web service Face++ provides a REST-api for uploading and recognition of images. amee-faceplusplus implements a custom SDK for communicating with the API using promises and file streams. The library was designed as a completely independent project in order to potentially be distributed as an open source project later on.

amee-easyup

A command-line tool which simplifies the manual managing of people and images on the Face++ server. Implemented with amee-faceplusplus.

amee-states

Implements an internal domain-specific language (DSL) for specifying a behavioral state machine. The focus of the library is to provide a natural and readable way to define a state machine. Designed with independent publication as an open source library in mind.

6.2. Prototype

Combining all the microservices creates our final prototype “Amee”. A complex state machine constructed in a modular hierarchy, where each microservice plays a role in the user experience. A simplified model of a session can be seen in figure 6.3.

The interaction is initialized by amee-doors, when the elevator door opens. The message is collected by amee-boombox which plays a random song. When

amee-cam detects a face, it takes a photo and sends a message to amee-recognizer and amee-controller. amee-recognizer posts the image to Face++ and awaits a response. amee-controller distributes a welcome message to amee-greeter and amee-voice. amee-greeter creates a new graphical item containing the message and displays it for the user while amee-voice reads the message out loud. amee-controller will post a stalling message every 2 seconds with small talk, to entertain/stall the user until a response from amee-recognizer is received. amee-greeter and amee-voice will continue to handle the messages.

When amee-recognizer gets a response, the response is collected by the amee-controller, which passes the message to amee-greeter and amee-voice. amee-controller then sends a question of whom to contact with possible responses, amee-greeter poses the question and amee-voice reads it aloud. amee-controller does not send any more messages but awaits a response from amee-greeter. When the user has selected a response, the response is received by the controller. It confirms the message by sending a message back to amee-greeter and amee-voice. amee-controller distributes a message to amee-nodered which posts a message on slack to whoever it may concern.

amee-controller continues by showcasing a video while the person is waiting. At any point, anyone can respond or write a message through slack to the GUI. The message is detected by amee-nodered who forwards it via amee-controller to amee-greeter and amee-voice. amee-greeter creates a special window just for slack messages in the upper left corner that can be displayed at anytime.

amee-door registers when the door opens, which notifies to amee-controller that the user is probably leaving the lobby and posts a goodbye message to amee-greeter and amee-voice. If the system has been active for more than 20 seconds, but no messages has been sent or received, a kill-session message is distributed that closes down amee-greeter. Behind all these microservices the amee-admin service is collecting all messages and publicize it on an admin website. In parallel amee-info posts its website on the inner greeter screens.

Even though the system is complex, it is modifiable. With not to much effort it is possible to add new features, new microservices or create new behavior altogether. The final prototype has a minimal feature-set and should be considered more of a proof of concept than a final product ready for production.

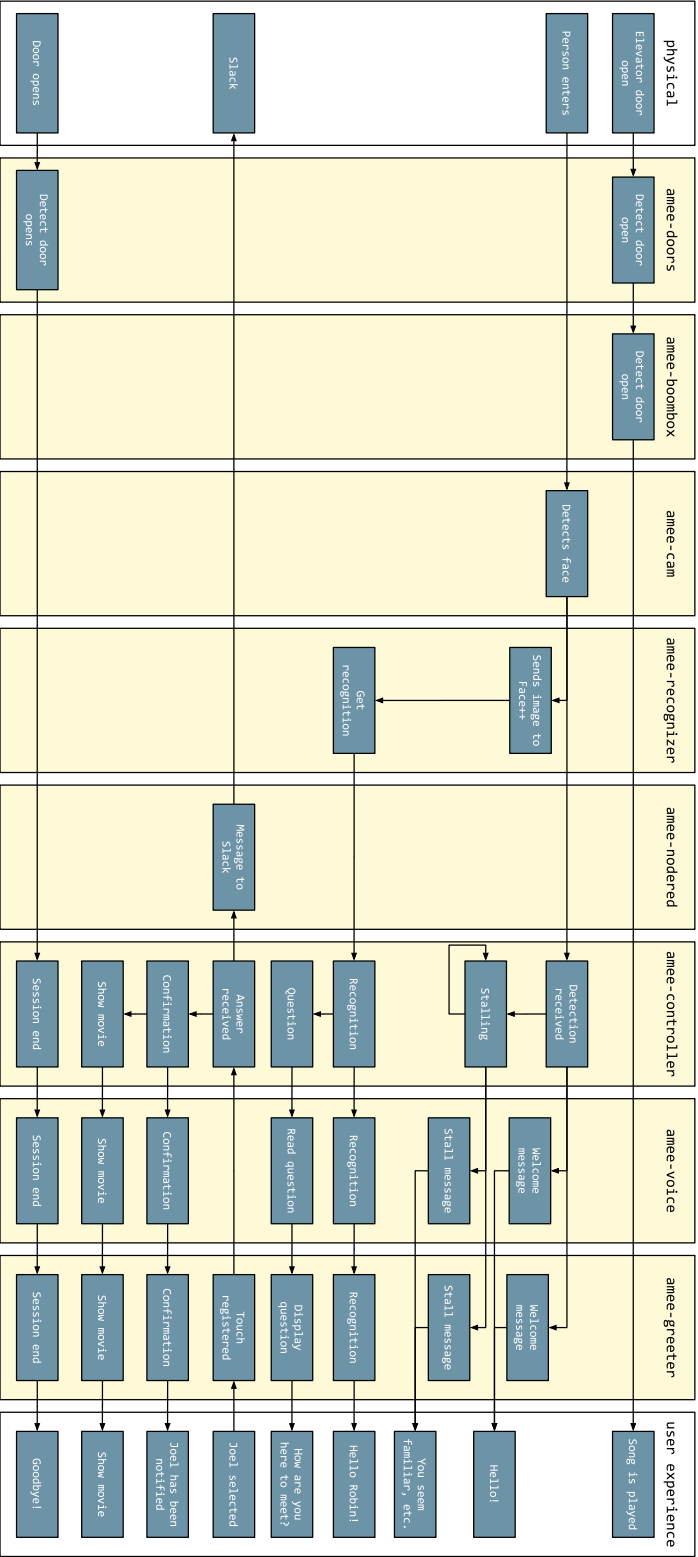


Figure 6.3.: An activity diagram of Amee

7. Discussion

This chapter will discuss the thesis work by first addressing each of the research question presented in the introduction (page 1). Then general thoughts on the execution of the project, followed up by reflections on ethics and future work.

7.1. Creating a welcoming atmosphere

How can we design and build a virtual receptionist that make visitors feel well received?

To achieve a welcoming atmosphere we have focused on the sensation of personal service and making arriving at the studio more fun. The result of this inquiry is Amee. While there is more that could have been done to expand on the concept, our interviews show that employees and visitors alike enjoy and appreciate even a basic implementation. Employees stated that they appreciated being greeted by Amee in the mornings and visitors were intrigued by the receptionist, enjoying her reaching out when they arrived to the studio. Everyone interviewed believed that Amee would make a good impression and be appreciated by clients coming to the studio. However, the interviews confirmed our own opinion, that the project is far from complete as compared to a proper receptionist. At best, Amee is at an early alpha stage.

7.2. Collecting and using contextual information

How can we collect contextual information and convey it through the virtual receptionist?

As stated in the definitions, we define contextual information as “any data with the potential to improve a design artifact’s user experience”. As such it can be information about pretty much anything. The challenge consists of differentiating between the data that is useful and that which is irrelevant. In developing Amee, we draw upon a number of sources of data.

As a way to kickstart the recognition engine, the amee-scraper tool helps collect information about name, nickname, gender, language preference and a varying number of pictures of a given person. This allows us to set up a basic database of information that can be used to recognize people and customize the receptionist's interaction accordingly. After this initial infusion of information we continually record photos of people entering the studio. While we haven't had time to fully automate the process of uploading and retraining the recognition service, this activity provides a good quality and homogeneous source of images of the people that have recurring business at the office. Lastly the above information and images are coupled with immediate area sensing in the form of face detection and physical sensors.

In the architecture implemented in the project, it is close to trivial to add more data sources. Simply mount a sensor somewhere and publish the data on a dedicated MQTT topic. The challenge is in combining different pieces of data into useful insights and in how to specify the conditions which should trigger a certain behavior. For example, when a person is in the lobby; the elevator door was opened most recently; Amee haven't yet been able to identify the person; and it was less than 1 second since the last message, she should say something to stall for time. Currently, these rules are implemented in the form of a state machine but its configuration has proved to quickly grow unwieldy, even though we only have a rather limited number of states. Even so, modeling a conversation as a state machine seems like the obvious approach. Perhaps it's simply the case that modeling a state machine lends itself better to a graphical representation rather than a written DSL.

7.3. Users feelings about privacy invasion

How will users feel about the digital agent knowing their personal information?

As shown in section 2.1, data privacy is important to most people and even considered a basic human right. It is, however, one of the rights people seems to be likely not to fight for when violated. Due to this paradoxical attitude to privacy, we were unsure how people would react to Amee. Perhaps they would view Amee as just another convenient service, or would they be alarmed by the fact that it knew their name and face?

The interviews, that was carried out towards the end of the project, indicated that people tend to accept that Amee knows their face without much difficulty. Even the hypothetical scenario where Amee was instead put in a convenience

store, which we expected to be met with more anxiety, didn't seem to worry them much.

All interviewees reported that they thought privacy was an important issue, though they had different approaches in addressing the issue. Several stated that to them, the cause was already lost and that it did not seem to matter anymore if additional information was released. A few commented that they actively work towards better privacy by, for example, restricting the content they share online or by disabling gps-tracking unless needed.

Why are not people more upset? We suspect that a combination of factors allowed the interviewees to easily accept Amee:

Perceived privacy: The small lobby where the visitor encounter Amee is relatively secluded, making the interaction feel more private.

Awareness: The interviewees seemed to be pretty aware about what information about them was public. As such, they were not overly surprised to find it known by Amee.

However there are a few factors that might have given us different responses. The only information that was presented to them were their own name. If we would have increased the exposure of personal content, e.g. showing them the picture used to identify them, show them Instagram images, Twitter feed, public Facebook information, status updates, etc. The responses could have been radically different. But, even then the chances of getting representative reactions might not have been possible with the selection group used, as they all were our acquaintances or colleagues.

7.4. Developing for the future

How does one build software that accommodates changing requirements?

There are many different theories and approaches one can try in order to build future-safe software. One of the most generally accepted premises in software engineering is modularization. Dividing software into distinct parts lets developers focus on one area at a time and isolates implementation details from other modules. The microservice architecture used in this project and described in section 6.1 is a good example of how far one can take modularization. Different modules (a.k.a. services) are not simply separated into packages within the same project, but run as completely independent programs with their own process, runtime, dependencies, implementation and physical location.

Besides the isolation of modules the topic-based publish-subscribe system used for communication between services results in a very flexible and fault-tolerant system (Birman and Joseph 1987). It is entirely possible, simple even, with our architecture to only use part of the system or repurpose a subset of modules for a different purpose than originally intended.

Empirically, the architecture has worked very well in decoupling different parts of the larger application and has proved itself to be easily adapted to the changes in requirements that arose during the thesis work. However, even though we are generally positive to this way of structuring an application, one should be aware that the benefits of this decoupling comes with some additional overhead to handle network issues and downed services (Lewis and Fowler 2014).

There are more ways to ensure the plasticity in a software project than ensuring a flexible architecture. In the project we have strived to adhere to common best practices of software development as deemed appropriate by us and Interaktionsbyrå. Having a good number of automated tests greatly helps in ensuring that changes and additions to modules do not break existing functionality. Also, while perhaps obvious to an IT professional, taking care to document each service in a README; naming variables and functions in a self-documenting way; and writing inline comments as necessary are all helpful to future developers trying to understand the codebase. Lastly, we have strived to use tools and frameworks of which other developers at Interaktionsbyrå are familiar - ensuring that they can take over the project without spending unnecessary time learning esoteric technologies.

7.5. Planning and Process

Over the course of the project many changes has been made to accommodate new information discovered as part of the work. Perhaps most importantly, as our understanding of the research area as well as the intention of Interaktionsbyrå developed, changes were made to the research question. Originally the question was:

How can we collect contextual information and leverage it in a digital receptionist in order to raise awareness about the issue of data privacy?

This evolved to the following main question:

How can we design and build a virtual receptionist that make visitors feel well received?

The final version focuses more on the welcoming user experience than the initial question and extracts the issue of managing contextual information and raising awareness about data privacy into separate subquestions - more closely reflecting their actual importance to the project.

Regarding the development, we were not able to implement all of our ideas. Some, due to lack of time. Some, due to them no longer being relevant to the project. Still, we are very pleased with the ecosystem that we were able to build. When programming started for real in the second half of the project, the work progressed very fast. Perhaps we should have started that work earlier. On the other hand, the speed with which we were able to implement features might very well be a direct consequence of having a well developed image of what needed to be done.

Lastly, while we produced a large number of sketches and rough prototypes it might have been a good idea to do some preliminary user testing early on to gain insight in what design was preferred. In these early stages however, major changes were an almost daily occurrence due to quick and dirty usability testing and discussion with employees. With so little stability, we would have risked wasting time on more formal testing of designs that would quickly grow obsolete. At the end of the project when the design had grown a bit more stable we did do some semi-formal testing. However, as the receptionist prototype is still rather basic, these tests are likely more suitable for further ideation rather than usability evaluation.

7.6. Ethical issues

Given the nature of the project we did inevitably face issues of privacy and personal integrity. While the employees at Interaktionsbyrån seem to universally trust the company to handle the collected data with discretion, some still felt uncomfortable about the project. To some extent, these concerns might have been aggravated by miscommunication in the early stages of the project. Perhaps we misjudged the sensitivity of the subject. Nevertheless, in collecting information we've only used that which is readily available publicly and which can be collected at the office. If users desire a higher level of privacy, they are encouraged to set a more restrictive policy in their respective social network accounts. Furthermore, the project naturally evolved to focus more on design and implementation issues rather than the issue of privacy awareness which was more heavily focused on in early stages of the project.

Another ethical issue encountered during the project was deciding on a gender for the virtual receptionist. Since we wanted the receptionist to feel like a person he/she should have a name. However, as to not reinforce stereotypical gender roles, we were hesitant to chose a female name - leaning more towards making the receptionist male, sexless or heterogeneous. The choice was to some degree made for us though when we added Google's text to speech service to the receptionist as it only provides a female voice. The voice quickly established the image of Amee as a woman at the office and changing it after the fact was deemed needlessly disruptive.

One, maybe unexpected, reflection about the virtual receptionist project concerns critical design. The offices of Interaktionsbyrån have two possible entrances. The first is the elevator leading into the lobby. The second one is a set of stair cases. The offices are on the 5th floor. Assuming that the Amee project is a success - are we causing people to use the elevator more often as a result of them getting a nicer user experience there? Are we then causing people to become lazier, helping them along the path toward obesity? Do we have a responsibility to enforce an equilibrium of 'fun' between the two entrances, or would that simply make us benevolent manipulators? Perhaps a more fatalistic approach is more politically correct. Granted, it might seem like a far fetched scenario, but the discussion above shows how any system could potentially have unexpected consequences not directly related to its purpose.

7.7. Future Work

Naturally, we would like to further expand on the prototype by adding more sensors and refine the conversation flow with more cases and customizations. We'd also like to simplify the administrative actions required to maintain the system by automating things like retraining and developing the administrative web interface to support creating people and uploading images. While long-term maintainability have been a focal point of development, we believe these additions would have a significant impact on whether the system is continued to be used after the end of the project.

Besides developing the prototype, a couple of interesting avenues of research we would like to explore have arisen through the work. One such area would be about how to model and build a conversational engine. Perhaps there is a model better suited to natural conversation than state machines? Maybe there are superior approaches to specifying state machines that would make them much more manageable as they grow.

Related to specifying conversation flows, could you build an artificial intelligence to handle the conversation in such a way that the programmer wouldn't have to specify it exactly? This topic has been revitalized in recent years with the introduction of services like Google Voice Search (Google 2011), Apple's "Siri" (Apple 2015) and even Wolfram alpha (Wolfram 2015). There are, however, much more that can be done in this area and the natural language processing which these services exemplify should be considered a related but not equivalent activity to *producing* natural language programmatically.

Yet another avenue of research is about image recognition. The Face++ service we use is arguably one of the best in the world but there are still many things that could be improved. One of the main issues in our case is how slow the recognition is since even though the image processing itself might be fast, we have to send the image-to-be-processed to Face++'s servers, either in China or North America. This introduces a significant lag which is very noticeable in the application. One way to deal with this would be to do the recognition on a local machine at the office. Even without a large server park to work on the processing it would still be significantly faster. However, when using Haar-like features, as OpenCV does (see section 4.2.6, p. 23), much processing time would be needed to retrain the recognizer - on the order of days with a moderate amount of images. Exploring how one could reduce or manage this significant processing would be very helpful to projects like this one.

A final interesting idea which one could look into further is in how to generalize the personalization showcased in this project for a more general audience. As more and more services are provided digitally the need to access basic information about an individual increases. Today, each company has to individually collect and manage this information. Perhaps there is room for a universal information service, potentially managed by the government or even a global organization, from which basic information about a person could be requested. It could even provide an image recognition service. Such a project raises a lot of questions about privacy and personal integrity but perhaps features like opt-in, encryption, fine-grained permission settings could make it practically and ethically feasible? What would be the implications for your everyday life if such a service existed?

8. Conclusion

During this project we have planned, designed and implemented a virtual receptionist and its architecture. Amee had a couple of goals; to give a personal greeting and welcome employees and guests when entering the studio, explore the possibilities and risks of data gathering/processing, and to create an architecture that is modular and easy to maintain.

At the most fundamental level the goals were met. The system greets everyone entering the studio, and if the individual is added in the database he or she is greeted by name. There is also the possibility to interact with Amee through touch input in the lobby and through Slack. The evaluation interviews indicated that users appreciate Amee and that the level of privacy invasion present in this application was considered harmless.

The system was built in a modular fashion meaning it is fairly easy to change parts of it without knowing the whole system. We quickly realized that the system would become more complex than originally thought. This meant that many planned features became put on hold. Naturally, time allowed only a small fraction of the extra features that were on the list. Nevertheless, the current level of fidelity has proved to be sufficient enough to answer the research questions, and to serve as a proof-of-concept.

To evaluate the prototype we had a number of interviews that confirmed our success in increasing the excitement of entering the lobby. The interviewees also indicated that Amee is likely to have a positive impact on visitors entering the studio. A surprising result was the impact of Amee on the company itself. There were discussions and ideas throughout the entire project of how Amee could expand out of the lobby and start having a broader set of responsibilities in the studio.

Finally, our thesis shows that with reasonable effort it is possible to create contextually aware applications. This, in combination with the ever advancing integration of smart devices into society (e.g. smart phones, smart homes, smart cars) the natural next step is to extend this into the realm of physical space in the shape of smart shops, smart walls, smart streets and so on. In order to create the even smarter devices and services of the future, an important challenge will be, in how to imbue them with context-awareness. We believe that in

order to realize this future, a vital step is in defining a universal protocol, which enables sharing of information between devices while still protecting our right to privacy.

Bibliography

- Aarts, Emile (2006). "Into ambient intelligence". In: *True Visions*, pp. 1–16.
- Abowd, Gregory D et al. (1999). "Towards a better understanding of context and context-awareness". In: *HUC '99: Proceedings of the 1st international symposium on Hand-held and Ubiquitous Computing*. Springer, pp. 304–307.
- Acquisti, Alessandro, Leslie K. John, and George Loewenstein (2013). "What is Privacy Worth?" In: *The Journal of Legal Studies* 42.2, pp. 249–274. ISSN: 00472530. DOI: 10.1086/671754.
- Alberti, Leon Battista (1435). *De pictura*. A Yale paperbound, Y-175. as translated by Spencer, J.R. 1966. Yale University Press. ISBN: 9780300000016. URL: <http://books.google.se/books?id=sVGZtXjRXPAC>.
- Apple (2015). *Apple - iOS - Siri*. URL: <http://www.apple.com/ios/siri/> (visited on 05/27/2015).
- Archer, L Bruce (1979). "Whatever became of design methodology". In: *Design Studies* 1.1, pp. 17–20. ISSN: 0142-694X.
- Arduino (2015). *Arduino*. URL: <http://www.arduino.cc> (visited on 05/22/2015).
- Argyris, Chris (1991). "Teaching smart people how to learn". In: *Harvard business review* 69.3.
- ARUP (2015). *Langham Place*. URL: http://www.arup.com/Projects/Langham_Place_HK.aspx (visited on 02/19/2015).
- Ashbacher, Charles (2003). "Test-Driven Development: By Example, by Kent Beck". English. In: *The Journal of Object Technology* 2.2, p. 203. DOI: 10.5381/jot.2003.2.2.r1.
- Augusto, Juan Carlos (2007). "Ambient intelligence: the confluence of ubiquitous/pervasive computing and artificial intelligence". In: *Intelligent Computing Everywhere*. Springer, pp. 213–234.
- MQTT Version 3.1.1 (2014). Tech. rep. OASIS Standard. URL: <http://eclipse.org/ponte/> (visited on 02/16/2015).
- Bazire, Mary and Patrick Brézillon (2005). "Understanding context before using it". In: *Modeling and using context*. Springer, pp. 29–40.
- Belhumeur, Peter N., João P Hespanha, and David Kriegman (1997). "Eigenfaces vs. fisherfaces: Recognition using class specific linear projection". In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 19.7, pp. 711–720.
- Berg, Bruce Lawrence, Howard Lune, and Howard Lune (2004). *Qualitative research methods for the social sciences*. Vol. 5. Pearson Boston, MA.

- Birman, K. and T. Joseph (1987). "Exploiting Virtual Synchrony in Distributed Systems". In: *Proceedings of the Eleventh ACM Symposium on Operating Systems Principles*. SOSP '87. Austin, Texas, USA: ACM, pp. 123–138. ISBN: 0-89791-242-X. DOI: 10.1145/41457.37515.
- Bocoup (2015). *Johnny-Five*. URL: <http://johnny-five.io> (visited on 05/22/2015).
- Brunch (2015). *Brunch Homepage*. URL: <http://brunch.io/> (visited on 02/16/2015).
- Cerejo, Lyndon (2015). *Design Better And Faster With Rapid Prototyping*. URL: <http://www.smashingmagazine.com/2010/06/16/design-better-faster-with-rapid-prototyping/>.
- Chai (2015). *Chai Assertion Library*. URL: <http://chaijs.com> (visited on 05/27/2015).
- Chamorro-Premuzic, Tomas (2014). *How much do we care about our online privacy?* URL: <http://www.theguardian.com/media-network/media-network-blog/2014/nov/10/online-privacy-digital-trust-psychology> (visited on 06/02/2015).
- Chui (2015). *Chui, The World's Most Intelligent Doorbell*. URL: <http://www.getchui.com/> (visited on 02/13/2015).
- CoffeeScript (2015). *CoffeeScript*. URL: <http://coffeescript.org> (visited on 05/27/2015).
- Cohn, Mike (2004). *User stories applied: For agile software development*. Addison-Wesley Professional.
- Collina, Matteo (2015). *Mosca*. URL: <http://www.mosca.io> (visited on 05/22/2015).
- Company, The Qt (2015). *Qt*. URL: <http://www.qt.io/> (visited on 03/31/2015).
- Council of Europe (1950). *European Convention on Human Rights*. English.
- Cross, Nigel (2006). *Designerly ways of knowing*. Springer. ISBN: 978-1-84628-300-0.
- Ducatel, Ken et al. (2001). "Scenarios for ambient intelligence in 2010". In:
- Eventboard (2015). *LobbyConnect, Digital Office Receptionist*. URL: <http://eventboard.io/products/lobbyconnect> (visited on 02/12/2015).
- Eventbrite (2015). *Eventbrite*. URL: <http://www.eventbrite.com> (visited on 05/22/2015).
- Express (2015). *Express*. URL: <http://expressjs.com> (visited on 05/22/2015).
- Git (2015). *Git –distributed-is-the-new-centralized*. URL: <http://git-scm.com> (visited on 05/27/2015).
- Google (2011). *Google Voice Search Official Announcement*. URL: <https://www.google.com/search/about/> (visited on 05/27/2015).
- (2015a). *AngularJS*. URL: <http://angularjs.org> (visited on 05/22/2015).
- (2015b). *Translate API*. URL: <https://cloud.google.com/translate/docs> (visited on 05/22/2015).
- Grau, Oliver (2004). *Virtual Art: from illusion to immersion*. MIT press.
- Greengard, Samuel (2012). "On the Digital Trail". In: *Commun. ACM* 55.11, pp. 19–21. ISSN: 0001-0782. DOI: 10.1145/2366316.2366323. URL: <http://doi.acm.org/10.1145/2366316.2366323>.
- Harrison, Steve and Paul Dourish (1996). "Re-place-ing space: the roles of place and space in collaborative systems". In: *Proceedings of the 1996 ACM conference on Computer supported cooperative work*. ACM, pp. 67–76.
- Holowaychuk, TJ (2011). *Mocha - simple, flexible, fun*. URL: <http://mochajs.org> (visited on 05/27/2015).

- Huang, Gary B et al. (2007). *Labeled faces in the wild: A database for studying face recognition in unconstrained environments*. Tech. rep. Technical Report 07-49, University of Massachusetts, Amherst.
- Huizinga, Dorota and Adam Kolawa (2007). *Automated defect prevention: best practices in software management*. John Wiley & Sons. ISBN: 0-470-04212-5.
- Humble, Jez, Chris Read, and Dan North (2006). "The Deployment Production Line". In: *Proceedings of the Conference on AGILE 2006*. AGILE '06. Washington, DC, USA: IEEE Computer Society, pp. 113–118. ISBN: 0-7695-2562-8. DOI: 10.1109/AGILE.2006.53.
- IBM Emerging Technology (2015). *Node-RED*. URL: <http://nodered.org> (visited on 05/22/2015).
- Ishii, Hiroshi and Brygg Ullmer (1997). "Tangible Bits: Towards Seamless Interfaces Between People, Bits and Atoms". In: *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*. CHI '97. Atlanta, Georgia, USA: ACM, pp. 234–241. ISBN: 0-89791-802-9. DOI: 10.1145/258549.258715. URL: <http://doi.acm.org/10.1145/258549.258715>.
- Itseez (2015). *OpenCV*. URL: <http://opencv.org/> (visited on 05/18/2015).
- Jade (2015). *Jade - Node template engine*. URL: <http://jade-lang.com> (visited on 05/27/2015).
- Joyent (2015). *Node.js Homepage*. URL: <http://nodejs.org/> (visited on 02/16/2015).
- Kumar, N. et al. (2009). "Attribute and simile classifiers for face verification". In: *Computer Vision, 2009 IEEE 12th International Conference on*, pp. 365–372. DOI: 10.1109/ICCV.2009.5459250.
- Laseau, Paul (1980). *Graphic thinking for architects and designers*. English. New York: Van Nostrand Reinhold Company. ISBN: 0442-2569.
- Lewis, James and Martin Fowler (2014). *Microservices*. URL: <http://martinfowler.com/articles/microservices.html> (visited on 05/21/2015).
- Ltd., Canonical (2015). *the leading OS for PC, tablet, phone and cloud | Ubuntu*. URL: <http://www.ubuntu.com> (visited on 05/27/2015).
- Manovich, Lev (2006). "The poetics of augmented space". In: *Visual Communication* 5.2, pp. 219–240.
- Megvii, Inc. (2015). *Face++*. URL: <http://www.faceplusplus.com> (visited on 05/22/2015).
- Mills, Gordon H. (1976). *Hamlet's castle : the study of literature as a social experience / by Gordon Mills*. English. University of Texas Press Austin, p. 362. ISBN: 0292730055.
- opencv-dev-team (2014). *FaceRecognizer - Face Recognition with OpenCV*. URL: <http://docs.opencv.org/trunk/modules/contrib/doc/facerec/> (visited on 02/11/2015).
- OpenIDEO (2011). *7 Tips on Better Brainstorming*. URL: <http://openideo.com/blog/seven-tips-on-better-brainstorming> (visited on 03/10/2015).
- Poitras, Laura. (Director) (2014). *Citizenfour*. Praxis Films.
- Privacy International (2010). *European Privacy and Human Rights*. English.
- Python Software Foundation (2015). *Python*. URL: <http://www.python.org> (visited on 05/22/2015).

- Rosenberger, Larry, John Nash, and Ann Graham (2009). "Blazing the new digital trail". In: *Business Strategy Review* 20.1, pp. 89–94. ISSN: 1467-8616. DOI: 10.1111/j.1467-8616.2009.00587.x. URL: <http://dx.doi.org/10.1111/j.1467-8616.2009.00587.x>.
- SIG, Bluetooth (2015). *Bluetooth®*. URL: <http://www.bluetooth.com/Pages/Bluetooth-Home.aspx> (visited on 05/29/2015).
- Slack Technologies, Inc. (2015). *Slack*. URL: <http://slack.com> (visited on 05/22/2015).
- Stutzman, Fred, Ralph Gross, and Alessandro Acquisti (2013). "Silent listeners: The evolution of privacy and disclosure on facebook". In: *Journal of privacy and confidentiality* 4.2, p. 2.
- Stylus (2015). *Stylus - Expressive, dynamic, robust CSS*. URL: <https://learnboost.github.io/stylus/> (visited on 05/27/2015).
- Sutherland, Ivan E (1965). "The ultimate display". In: *Multimedia: From Wagner to virtual reality*.
- Taigman, Y. et al. (2014). "DeepFace: Closing the Gap to Human-Level Performance in Face Verification". In: *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pp. 1701–1708. DOI: 10.1109/CVPR.2014.220.
- Tuan, Yi-Fu (1977). *Space and place: The perspective of experience*. U of Minnesota Press.
- Turing, Alan M (1950). "Computing machinery and intelligence". In: *Mind*, pp. 433–460.
- UN General Assembly (1948). *Universal Declaration of Human Rights*. English.
- Venturi, Robert, Denise Scott Brown, and Steven Izenour (1972). "Learning from Las Vegas". In:
- Wang, Raymond (2004). *Langham Place offices to roll out next month*. URL: http://web.archive.org/web/20040820014709/http://www.thestandard.com.hk/thestandard/news_detail_frame.cfm?articleid=48588&intcatid=1 (visited on 02/19/2015).
- Weiser, Mark (1991). "The computer for the 21st century". In: *Scientific american* 265.3, pp. 94–104.
- (1994). "The world is not a desktop". In: *interactions* 1.1, pp. 7–8.
- Weiss, Aaron (2009). "A Digital Trail is Forever". In: *netWorker* 13.2, pp. 14–19. ISSN: 1091-3556. DOI: 10.1145/1540392.1540397. URL: <http://doi.acm.org/10.1145/1540392.1540397>.
- Wilson, Phillip Ian and John Fernandez (2006). "Facial Feature Detection Using Haar Classifiers". In: *J. Comput. Sci. Coll.* 21.4, pp. 127–133. ISSN: 1937-4771. URL: <http://dl.acm.org/citation.cfm?id=1127389.1127416>.
- Wolfram (2015). *Wolfram Natural Language Understanding System™*. URL: <http://www.wolfram.com/natural-language-understanding/> (visited on 05/27/2015).
- Zeit (2011). *Betrayed by our own data*. URL: <http://www.zeit.de/digital/datenschutz/2011-03/data-protection-malte-spitz> (visited on 02/10/2015).

Appendices

A. Definitions

Digital Receptionist A digital agent that welcomes and manages the flow of visitors to a location.

User Experience The sum of the users impressions during a usage scenario. In this project, special attention will be given to the aspects of impact (making an impression on the user) and transience (natural and thoughtless interaction).

Contextual information Any data with the potential to improve a design artifact's user experience.

Face Detection Detecting the presence of a face in an image.

Face Recognition Identifying a person using his or her face.

Inlet An artifact that collects data for a central database.

Outlet A design artifact that makes use of the central database.

Broker An MQTT server providing access to an underlying database.

Client (MQTT) A client can request data from a broker or subscribe to updates.

B. User Stories

As an admin, I want to block offensive content.

As an admin, I want to manage people in the system.

As an admin, I want to minimize manual labor.

As an employee, I want to get through the door as quickly as possible.

As an employee, I want my work place to be fun and creative.

As an employee, I want to have lunch together with my colleagues.

As an employee, I want look good in front of colleagues, clients and bosses.

As an employee, I want to know what time it is where colleagues and clients.

As an employee, I want to know when colleagues birthdays are.

As an employee, I want to know when colleagues will arrive.

As an employee, I want to know what appointments I have.

As an employee, I want to customize my user experience.

As an employee, I want to be informed of upcoming events.

As the owner, I want to showcase the company for the visitors.

As the owner, I want to impress visitors.

As a user, I want to separate private life from professional life.

As a user, I want to be greeted in my own language.

As a user, I want to get to places quickly.

As a user, I want to pass time waiting for the elevator.

As a user, I want my birthday to be remembered.

As a user, I want to interact effortlessly with the receptionist.

As a user, I want to get a weather forecast.

As a user, I want to know where people are in the office.

As a user, I want to know when I'll arrive at my destination.

As a user, I want to know where to get lunch.

As a visitor, I want to feel welcome.

As a visitor, I want to quickly get in touch with my contact.

As a visitor, I want to know if someone is on their way to open the door.

As a visitor, I want to make sure I look good.

C. Interviews

C.1. Employees at Interaktionsbyrån

C.1.1. Employee 1

Have you tried Amee recently? Yes, every day when I enter!

When did you learn about the project? Joel told me. Before you were here.

Before the project started, do you remember your initial thoughts on what the virtual reception would be like? Not really. I expected it to do something cool.

Have you've experienced any similar system before? No. I don't think it's allowed. *laughter*. It's pretty unique.

What do you think of the system now? It's nice! It needs more functionality but it works as a proof-of-concept. Sometimes it recognizes me as someone else. It gets confused when there are a lot of people, You don't know who its recognizing.

How does it affect you in your everyday? No, But for the moment I don't need the help it provides.

Do you enjoy it or does it just annoy you? No it's nice. I enjoy it in the mornings! Sometimes the text-to-speech is kind of weird, when it tries to say foreign words in an american accent.

Can you describe how you felt when it recognized you for the first time?
How is it now? It was nice. Felt good/cool.

Do you perceive the system and project to be innovative or old news? I haven't seen any public systems. Using it as a more obvious receptionist it would probably be cooler. It's usefulness is limited for employees. For welcoming people it's fine.

Imagine the system being somewhere else, do you think it would affect your perception if a similar system would exist in a store, or at home? I don't know. I'd probably think it's nice. I don't really care much about privacy.

It's a lost cause anyway. It could probably do more, like in a sci-fi movie, if it was better at recognizing you.

Do you think that the system affects or can affect customers and clients of Interaktionsbyrån? I think they'll like it. As long as it provides something for them. Especially as a new gimmick they haven't seen before. You could probably work more on the content after being greeted.

Do have any more comment you would like to add? It's not obvious that there's a touch screen and you should have a game!

C.1.2. Employee 2

Have you tried amee recently? Yep!

When did you learn about the project? Sometime before you started. Tried to stay out of it.

Have you've experienced any similar system before? Not like this, many companies have automatic entry-systems, were you log-in and then get a guest pass.

What do you think of the system now? It does not fill any purpose at the moment.

How does it affect you in your everyday? No, effect, but there is potential.

Can you describe how you felt when it recognized you for the first time?
How is it now? It was fun. It did not surprise me as I knew it was coming.

Do you perceive the system and project to be innovative or old news? It feels quite innovative, but for us employees enters the door to quickly for the product to have any meaning. It could however be good for guests.

Imagine the system being somewhere else, do you think it would affect your perception if a similar system would exist in a store, or at home?
Like the food store? It could have a purpose, for example when you have self-scanning. If it new it was me it could be prepared for me to pay by card as I always do that.

Do you have personal policy on integrity or any thoughts on the matter?
All the photos are sent to the US, this means the NSA have the knowledge of everyone that enters here. It bothers me. It is okay here at Interaktionsbyrån. But I do think that integrity is an important issue.

Do you think that the system affects or can affect customers and clients of Interaktionsbyrån? Hopefully it will be in a good way, that it textbfesises

that were an innovative company. Sure, some people might not like to have a camera stuck in their face. But the clients that work with us are probably not that sensitive.

Do have any more comment you would like to add? It is pretty good at recognizing me. Never gotten mer wrong.

C.1.3. Employee 3

Have you tried amee recently? Yes, I've tried some everything except mentioning Amee on slack.

When did you learn about the project? Fairly early. Even before it started. Some ideas were floating around.

Before the project started, do you remember your initial thoughts on what the virtual reception would be like? Not really. Everything was very general. No expectations.

Have you've experienced any similar system before? No. Nothing like that.

What do you think of the system now? It's always fun to be recognized. With voice you really captured the attention. Before, people kind of ignored it. I can imagine the amount of work you've put into it.

How does it affect you in your everyday? I can see some stuff in the future, but not really right now.

Do you enjoy it or does it just annoy you? It's fun. If it's boring I just walk past.

How does the recognition work for you? How often does it succeed? About 80%. Less after I shaved.

Can you describe how you felt when it recognized you for the first time? How is it now? It was pretty fun. Especially when it worked consecutively.

Do you perceive the system and project to be innovative or old news? It's not old news. I think it has potential. Even if face recognition is fairly easy nowadays, it's still novel. It still surprises people. Often similar things are simply static, and this feels more like a person.

Imagine the system being somewhere else, do you think it would affect your perception if a similar system would exist in a store, or at home? I would probably be a bit more creeped out, since it's a more public place. Here

it's like meeting a friend. At a shop it would be more like a stranger knowing my name. If I'm a regular, like at a bar or something, it'd feel slightly better.

Do you have personal policy on integrity or any thoughts on the matter? I try to use stuff that is paid and vpn and stuff. Though sometimes you have to use services which uses data collection, such as Google and similar.

Do you think that the system affects or can affect customers and clients of Interaktionsbyrå? I haven't seen any outsider be exposed to it, so I don't really know. I would think it would be a surprise or a novelty. Some might be creeped out, but most will be ok. As this is a trusted place it would be fine.

Do have any more comment you would like to add? Nothing really.

C.1.4. first time user 1

What did you think? Very nice, and it worked! If it would not have worked it would not have been that impressive.

Can you compare the device to anything you have experienced before? No, The closest is for example Telia's phone support. "press 1 to...".

What did you think when it said your name? It was cool. Especially because I have never been here before.

Did you believe it was faked? No, I didn't.

Do you think you would have had a different experience if you would have been a client? I still think I would have thought it was cool. But I imagine that you are more in a hurry that you would not like to wait for Amee. The first probably still has a wow-factor, but the 10th time you probably don't care that much anymore.

You tried to speak to Amee, but as you noticed it did not work. How did you experience the level intuitiveness? Yes, I thought, because it spoke to me, that it would be a two way conversation. But I realized that it could not and then the touch became a natural next step.

Do you use social media, e.g. Facebook, twitter? Yes, Facebook, LinkedIn.

How much of your life do you share on these sites? Almost nothing. It does not interest me. It has nothing to do with data integrity.

Do you have a personal stance on data integrity? Kind of. There is certainly a line. But in this case I did even think of private issues.

Do you think you would have had a different opinion if Amee was in you local super market? I don't think I would have thought so much about it. Probably more about if the system were broadcasting personal information about me so others could hear. Here, the room is rather controlled so it does not matter.

Do you have anything to add? You should make the buttons more clearer. Maybe have an express mode for people that come often? Automated notification if there is a booked meeting.

C.1.5. first time user 2

What did you think? It started to talking to me, "what!?" Sometimes when you enter an office you don't know what to do. It was really nice that it took charge. I did not notice that it got my name wrong when I got the the question of who to contact. unfortunately I had already pushed the the doorbell by then. I did not understand where it was going with the conversation.

Can you compare the device to anything you have experienced before? Kind of, only machines where you enter your name and registered your arrival.

Do you think you would have had a different experience if you would have been a client? I still think I would have thought it was a nice thing. It feels new!

You tried to speak to Amee, but as you noticed it did not work. How did you experience the level inuitiveness? It felt easy. But reality is still around you which made me to loose focus, especially when it was just stalling for a long time. That's when the doorbell wins.

Do you use social media, e.g. Facebook, twitter? Yes, Facebook, Instagram.

How much of your life do you share on these sites? I don't share that much information about my life. Use it mostly for communication and arrange events. I also use Instagram some, but just because I like the art of photography rather than documenting me.

What do you think you would have thought of Amee if it was in you local super market? I always think those things are a little bit uncomfortable. But still it is good that you are informed of how much information there is about you.

Do you have a personal stance on data integrity? I usually try to keep an eye of what is shared about me. But there is a line when you don't have the

energy to keep track on everything. Like Google, it is very flexible, I use it even if Google gets all my information, and the negative effect seems to be invisible.

Do you have anything to add? It worked very well, considering how confused I was.

D. User Scenarios

The wind howls outside when the lonely wanderer enters the elevator. As the door closes, noises of car horns and tram squeals softly abandons the auditory space, replaced by the solitude of silence. The wanderer is tiered, the usual morning traffic, rain and total darkness gets even the strongest soldier, it has been witnessed many times before. The button is pressed and the journey to the 5th floor has started. Except for soothing sound of the mechanical movement and the electrical engine, the small box is silent. It gives the wanderer time to reflect, with too few hours of sleep it is hard to fight the absence of inspiration. "...life sucks..." the wanderer mumbles while changing his gaze from his feet to the roof of the elevator, sweeping by the mirror, reflecting the horrible sight that is the wanderers own self. The wanderer flinches. What is that? At the horizon of his hearing the song approaches in a steady pace. Even though the discretion which the song is played, the wanderer recognizes the music immediately. "Ride of the Valkyr" by Wagner. As the elevator approaches it's final destination the absence of inspiration fades. Moments later the bad morning, the annoying people at the gym last night, disappears. The mixed peculiar feeling of awesomeness and rage fills the wanderers whole body all the way out into his fingertips. *Ding* The door opens, a screen flashes once and the sound of thunder fills the small hallway with a determined bass shaking his inner core. The dark screens transforms and starts to write in real time. "Good morning"... "Foobar" ... "You look awful this morning!" ... "21 % beauty is the worst that I have ever recorded..." ... "Congratulations! You must be very proud :) I'll upload a picture of you on your facebook." ... "Maybe you should go to the bathroom take a shower and I'll prepare some coffee in 10 min." ... "Sounds good?" The wanderer nods discreetly, he has not really made up his mind on what his feeling towards Amee is yet. For some reason he finds Amee's douchebag-approach be kind of funny. "Good! I'll do that... and by the way welcome again... This is somewhat of a jubilee as it is the 200th entry here!" ... "Here this is for you!" A very sad flower is displayed on the entire screen. The wanderer just enters without a second glance... "Jerk!"