Short summary

In this thesis we develop an online cash register application for restaurants and events using a case study proposed by Faros, a company part of the Cronos group. This application is a digital version of cash registers found in stores and restaurants. Customers can use their own mobile devices to get an overview of the products available at a location and place orders. Employees use computers or their own devices to receive and process these orders. The application is developed with distribution in mind and functions as a web application running in the cloud.

The core aspect of this project is exploring Java web technologies and analysing what features they offer. We research frameworks and libraries suitable for prototyping the case study, and define constraints and other requirements with our coaches at Faros. We focus specifically on distributed applications and the micro-services architecture. Multinational companies such as Netflix, Amazon and eBay are already running on micro-services and it is becoming more popular in the web development world. We explore the advantages and disadvantages of this kind of architecture by developing the online cash register application. First as a single monolithic application and then rebuilt into a micro-service system which can run as several separate modules on multiple servers.

At the end of this project we should have an excellent understanding of the technologies necessary to create an effective and scalable web application. By developing the online cash register case study, we have increased our general knowledge of web design and software development. This should improve our ability to select the most appropriate tools in the development of future applications. It also clarified the difficulties and advantages of implementing a micro-services system.

Extended technical summary

In this thesis we develop an online cash register application for restaurants and events using a case study proposed by Faros, a company part of the Cronos group. This application is a digital version of cash registers found in stores and restaurants. Customers can use their own mobile devices to get an overview of the products available at a location and place orders. Employees use computers or their own devices to receive and process these orders. This case study follows the standard procedures used in application development. We do market research, perform a use case analysis, determine the necessary and optional features, create navigational models and UML diagrams where needed. The application is developed with distribution in mind and functions as a web application running on multiple separate servers.

The core aspect of this project is exploring Java web technologies and analysing what features they offer. We research frameworks and libraries suitable for prototyping the case study, and define constraints and other requirements with our coaches at Faros. This will include many technologies related to enterprise java development. A framework such as Spring or JSF is used to serve HTML pages from an application server to clients. For storing data, we look at both relational and document-based databases such as MySQL, OracleDB and MongoDB. Where beneficial, Object mapping tools are used. Security systems for authorisation and authentication are also discussed. Finally, we also look at front-end frameworks and libraries such as AngularJS and React and consider the advantages they offer.

The primary focus however, goes to the back-end architecture of the application. We look specifically at applications hosted on cloud platform like Amazon AWS, Microsoft Azure, IBM cloud and more; determining the appropriate way to develop applications which utilize the advantages of distributed platforms. For this we will look at the micro-services architecture and the advantages it offers for scaling a system to serve a large userbase. Multinational companies such as Netflix, Amazon and eBay are already running on micro-services and it is becoming more popular in the web development world. By developing the online cash register application with this architecture, we analyse the advantages and disadvantages of using micro-services. First a single monolithic application is created using development techniques familiar to us. This application will be a monolithic program that includes an application server, all the required mappings and classes, plus a connection to a single relational database. After successfully building a prototype application that implements the features specified in the case study, we restructure the project to a micro-service architecture. Business code for the services are extracted from existing modules and the application server will become a separate module. Each server gets its own database, using the technology most suited to data stored by the corresponding service. Additional services are created to support this micro-service architecture such as a Discovery service and a gateway. The final result is an effective implementation of the Api gateway design pattern.

At the end of this project we should have an excellent understanding of the technologies necessary to create an effective and scalable web application. Following the development

process above, we will get insight into the advantages of a micro-services architecture, but also the problems that come paired with the development of such a system. Especially when migrating to a micro-services architecture from an existing application. By developing the online cash register case study, we have increased our general knowledge of web design and Java development. This should improve our ability to select the most appropriate tools in the development of future applications.