Kickstarting Application Development with Enterprise Java
A Spring Boot Case Study

Lennart Cockx
Guangming Luo

April 26, 2017

Contents

Abstract

In this report we will explore various Java technologies and analyse what features they offer. The focus of this project is to find which frameworks are most suitable for quick prototyping of a business application. Constraints and other requirements will be defined and will then be applied to a specific case study.

1 Project Objective

1.1 Proposal

This master thesis is commissioned by Faros, a Cronos group company specialized in the development of Java web applications and primarily focused on designing Rich Internet Applications. Development within Faros is done mainly with Vaadin, Spring, HTML5, ExtJS, JavaScript, JSP, JSF, NoSQL and more. This means that during this thesis, special attention is given to these technologies in regards to design choices. Naturally, comparisons with other possible frameworks will still be done and valid arguments will be set forward to confirm these decisions.

The following case study was proposed by Faros: designing an online cash register that can be used in restaurants or at events. Existing cash register systems are expensive and for smaller establishments or temporary events, this investment is not worth it. The idea is to use existing devices from employees and customers to handle the ordering and payment process.

This project is an opportunity for us to learn about the frameworks and technologies used for the development of rich, interactive web applications. Additionally, we want to explore the best methods to quickly build a web application from scratch. Some key aspects that we will look out for are: Documentation availability, speed of new project setup, customizability, integration with other frameworks and security features. When given the corresponding documentation, it should be possible to reproduce this application

1.2 Concept

At the start of the project we discuss the key aspects of this project with our coaches at Faros. Based on the requirements provided by them, combined with our own interests, we will define the general field we will be working in.

After we have a good overview of what aspects must be included in this project (or should be avoided), we can start deciding core features that will determine which technologies will be appropriate for this project. These choices are based on the aspects we think are needed for developing a web application based on the aforementioned requirements. This should give us a list of possible systems and frameworks that could potentially be used during this project.

Now that we have a list of possible systems, we can start to think about the case study we want to develop. Once we know the general outline of the project, we can begin selecting the the appropriate tools and frameworks that fit this case study. This will shorten the list to technologies specifically applicable to this project. For certain frameworks it will be difficult to obtain a shorter list, as they might all comply with the requirements we set. In this case we will make a selection of items to keep the scope of the project realistic. Usually the most popular frameworks will be selected. If a deviating set is chosen this will be elaborated on in the corresponding chapter.

Now that the tools are known, the standard procedure for application development will be followed. Relevant documentation will be constructed including a use case analysis, feature description (Nice-to-have versus Must-have), navigational models, hierarchical task analysis, UML diagrams and market research. After this we can start working on getting familiar with the chosen enterprise Java frameworks.

As soon as we are sufficiently proficient with our chosen tools, we will start working on a case study where we use the knowledge we obtained to build a functional prototype for a realistic use case scenario. During the project we might come to the conclusion that certain choices we made were suboptimal or simply do not allow further progression. In such cases, we will adjust our trajectory and change to the appropriate tools. Any time such a decision is made we will mention it in this report and provide the matching argumentation as to why we made the switch.

1.3 Project Requirements

1.3.1 Constraints

1.3.2 Design focus

2 Enterprise Java Research

2.1 Current Landscape

2.2 Comparison of technologies

2.2.1 General framework

2.2.2 Persistence providers

2.2.3 Security

2.2.4 Front-end and Styling

3 Case study: Online cash register

3.1 Introduction

The goal of this case study is to create a service that can be used at events and in restaurants to process orders made by customers, relay these orders to the waiters and chefs and handle the payment. This service should include an application that clients can use to view the menu and order items from it. Waiters also have their own application that can place and acknowledge orders. These orders will be send to a central system that manages all the required information.

3.2 Use Cases

In this case study we can identify several distinct actors that will interact differently with the application.

Customer The customer is the primary user of this application. They access the menu of a specified restaurant or event. They can place orders, either via the application itself or from a waiter with their own device. Finally, they can pay for the orders that they placed. Again, either via the application or a waiter.

Waiter Like the customers, the waiter can access the menu of their restaurant and place orders for their clients. In addition to this they can display an overview of the currently queued orders, with the corresponding table numbers or location. They will also handle payments from customers through the application, or accept payment in other forms of currency.

Chef The chef can access similar functions as the waiter.

Manager

3.3 Must-haves and nice-to haves

3.4 Technology choices

3.4.1 Application type

3.4.2 Framework choices

3.5 Design Choices

3.5.1 Client

3.5.2 Employee

3.5.3 Manager

3.5.4 Remarks and challenges

3.6 Major issues and their solutions

blablabla [2] [1]

References

[1] Alex Antonov. *Spring Boot Cookbook*. Packt Publishing Ltd, 2015.

[2] George D. Greenwade. The Comprehensive Tex Archive Network (CTAN). *TUGBoat*, 14(3):342–351, 1993.