**Club Manager Design Document**

Cristin Rusnac 3616908

ITS, Fontys University of Applied Science

Table of Contents

[Abstract 3](#_Toc73547100)

[Product Backlog 4](#_Toc73547101)

[Burndown Chart 6](#_Toc73547102)

[Versions of the Solution 7](#_Toc73547103)

[Frontend Framework 8](#_Toc73547104)

[C4 model 11](#_Toc73547105)

[Entity Relation Diagram 16](#_Toc73547106)

Abstract

The concept of the project is to develop an application which is supposed to be a Training Manager for a soccer club. The purpose of this Full-Stack Web Application is to offer the User the possibility to Authenticate in the system, retrieve all the exercises of the training, add new exercises to the training, edit and delete them. Also view the calories per exercise added based on the exercise type and the duration of the exercise and as well retrieve the total amount of calories of the Training.

Product Backlog

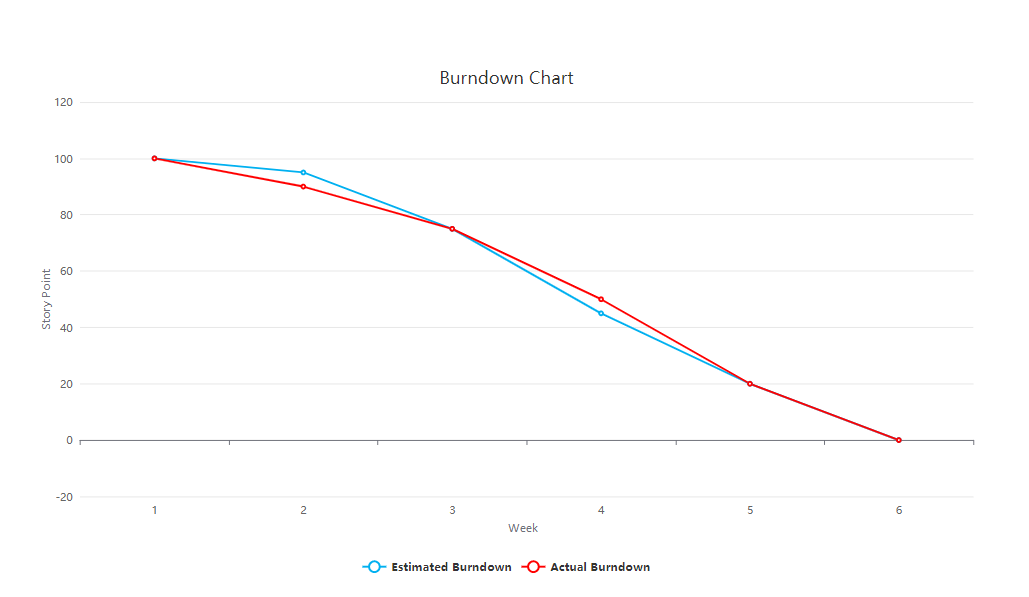
|  |  |  |
| --- | --- | --- |
| User Story | Prioritization | Estimation |
| As a User I can access the Login page and fill in my email and password to authenticate in the system.  Acceptance Criteria:   * The form page is accessible * It is possible to fill in the login credentials * When clicking the Login button the Authentication and Authorization process is working accordingly. | **60** | **15** |
| As a User I want to be able to access the training page in order to view the details of each exercise and the training as a whole.  Acceptance Criteria:   * There is a navigation bar with the appropriate links. * The information on the training page is viewed in a user friendly layout and is easy to understand | **100** | **10** |
| As a User I want to be able to add new exercises to the training in a appropriate form and display the newly added exercises on the training page.  Acceptance Criteria:   * The Exercise submission form is available through the navigation bar * You can fill data in the form and submit it | **90** | **15** |
| As a User I want to make changes in any of the exercises that are already part of the training.  Acceptance Criteria:   * There is an appropriate button next to each exercise on the training page to access the edit page of that specific exercise * You can make changes in the edit form and submit the changes. | **60** | **15** |
| As a User I want to be able to delete any exercise from the training.  Acceptance Criteria:   * There is an appropriate button next to each exercise on the training page that allows the user to delete that exercise * When you click the delete button of a specific exercise, the exercise is removed from the training | **70** | **5** |

|  |  |  |
| --- | --- | --- |
| As a User I want to be notified each time I am making any changes in the training (adding, deleting, editing exercises).  Acceptance Criteria:   * When a change is made there is a popup in the right top corner of the screen that notifies the user of the change he made | 40 | 10 |
| As a User I want the training to display the calories I will burn next to each exercise, based on the exercise type and the assigned duration for the exercise, and also the total amount of calories of the training.  Acceptance Criteria:   * There is a logic that calculates the number of the calories based on exercise type and duration assigned to it. * There is an appropriate field on the training page that displays the information regarding the calories next to each exercise. * There is a logic that calculates the total amount of calories of the training. | **50** | **25** |
| As a User I want to logout of the system.  Acceptance Criteria:   * There is an appropriate button on the navigation bar that will logout the user and forward him to a neutral page. | **60** | **5** |

Prioritization: 0-100 points

Estimation: Total of 100 points

Burndown Chart



Versions of the Solution

|  |  |
| --- | --- |
| **Version** | **Changes** |
| **0.1** | First Setup of the Backend with Models and Controllers |
| **0.2** | Addition of Mock Data to the Backend |
| **0.3** | First setup of CRUD operations of the Backend like GET and DELETE |
| **0.4** | Implementation of the POST operation |
| **0.5** | Development of a basic frontend with the GET operation |
| **0.6** | Added a submission form to the frontend and Navbar |
| **0.7** | Structured and separated the data on Frontend |
| **0.8** | Addition of the DELETE and PUT features to the frontend |
| **0.9** | Implementation of a Postgress Database for the backend |
| **1.0** | Implemented testing and sonarqube for backend |
| **1.1** | Implementation of React Redux on the Frontend |
| **1.2** | Implementation of Authentication and Authorization |
| **1.3** | Implementation of the calories determination |
| **1.4** | Implementation of the notifications for the Frontend |
| **1.5** | Implemented the total calories determination, testing, docker and setting pipelines |

Frontend Framework

React (also known as React.js or ReactJS) is an open-source, front end, JavaScript library for building user interfaces or UI components. It is maintained by Facebook and a community of individual developers and companies. React can be used as a base in the development of single-page or mobile applications. However, React is only concerned with state management and rendering that state to the DOM, so creating React applications usually requires the use of additional libraries for routing. React Router is an example of such a library. And also large and complicated applications are required to make AJAX calls to fetch or mutate data. You can use any library you like with React such as Axios, JQuery AJAX or the browser built-in window fetch. (Wikipedia, 2021)

Ease of learning

Both Angular and React require a more complex project setup with advanced developer tooling (like Webpack) to (realistically) get started. Though, bigger Vue projects (where you want to use the above-mentioned “Single File Component” syntax) also require a more complex project setup.

Creating projects with fitting setups is made easy though - you do have the Angular CLI for Angular projects and create-react-app for React projects. In case you need a more complex Vue project setup, you can easily get it with help of the Vue CLI. (Schwarzmüller, 2020)

React is definitely a bit more difficult to get into than Vue is yet simpler than Angular.

Performance

*Startup Performance*

Angular tends to be a bit bigger than Vue and React apps - with Vue then being a bit smaller than React.

For bigger apps, all three frameworks should produce roughly equally-sized code bundles - assuming you are using optimization techniques like “Lazy Loading” which all three frameworks support. (Schwarzmüller, 2020)

*Runtime Performance*

React is not just focusing on using all possible JavaScript tricks to detect and perform all necessary UI updates efficiently. Instead, it also utilizes techniques where it priorizies certain tasks to make the loaded page feel faster to the end user. It might, for example, handle user input with priority and delay the update of some text being output somewhere on the screen. Of course, when I write “delay” here, we’re typically only talking about (a few hundred) milliseconds. (Schwarzmüller, 2020)

Integration with Backend

React was designed to make it painless to create interactive UIs. Its state management is efficient and only updates components when your data changes. Component logic is written in JavaScript, which means you can keep state out of the DOM and create components that are encapsulated.

Developers like CRUD (create, read, update, and delete) apps because they show a lot of the base functionality that you need when creating an app. Once you have the basics of CRUD completed in an app, most of the client-server plumbing is finished, and you can move on to implementing the necessary business logic.

React is one of the easiest to integrate with a Java backend application, even if you search online you can find a lot of tutorials on full-stack applications specifically for React and Spring Boot combination. (Raible, 2018)

Spring Framework

The Spring Framework is an application framework and inversion of control container for the Java platform. The framework's core features can be used by any Java application, but there are extensions for building web applications on top of the Java EE (Enterprise Edition) platform. Although the framework does not impose any specific programming model, it has become popular in the Java community as an addition to the Enterprise JavaBeans (EJB) model. The Spring Framework is open source.

***Spring Boot***

Spring Boot is Spring's convention-over-configuration solution for creating stand-alone, production-grade Spring-based Applications that you can "just run". It is preconfigured with the Spring team's "opinionated view" of the best configuration and use of the Spring platform and third-party libraries so you can get started with minimum fuss. Most Spring Boot applications need very little Spring configuration.

**Features:**

* Create stand-alone Spring applications
* Embed Tomcat or Jetty directly (no need to deploy WAR files)
* Provide opinionated 'starter' Project Object Models (POMs) to simplify your configuration
* Automatically configure Spring whenever possible
* Provide production-ready features such as metrics, health checks and externalized configuration
* Absolutely no code generation and no requirement for XML configuration.