DESIGN DOCUMENT

"Cinema_Now"

Andreea Șindrilaru
Fontys University of Applied Sciences
S3-CB-01

Table of Contents

1. Introduction

1.1. Purpose of the document

The software design document is meant to provide a clear description of the design of the web application for the individual project of the 3rd semester. The document's purpose is to provide a clear understanding of what needs to be built and how it is expected to look like at the end. The design document will cover the frontend as well as the backend of the software, thus giving a clear insight of every aspect of the design of the application.

1.2. Document Overview

This document serves as a description of the functionality and software design for Cinema_Now application.

1.3. Definitions, Acronyms and Abbreviations

API : is the acronym for Application Programming Interface, which is a software intermediary that allows two applications to talk to each other. Each time you use an app like Facebook, send an instant message, or check the weather on your phone, you're using an API.

Interface: A software interface may refer to a wide range of different types of interfaces at different "levels": an operating system may interface with pieces of hardware. Applications or programs running on the operating system may need to interact via data streams, filters, and pipelines and in object-oriented programs, objects within an application may need to interact via methods.

Axios: is a promise-based HTTP Client for node.js and the browser. It is isomorphic (= it can run in the browser and node.js with the same codebase). On the server-side it uses the native node.js http module, while on the client (browser) it uses XMLHttpRequests.

OOP (Object-oriented programming) is a programming paradigm based on the concept of „objects", which can contain data and code: data in the form of fields (often known as attributes or *properties*), and code, in the form of procedures.

REST/CRUD means using the HTTP, GET, POST, PUT and DELETE operations to implement CRUD operations: - use of client-server architecture, stateless communication, etc.

C4 Architecture is an easy to learn, developer friendly approach to software architecture diagramming, assisting with communication inside/outside of software development/product teams.

Class is an extensible program-code-template for creating objects, providing initial values for state (member variable) and implementations of behavior (member functions or methods).

React (also known as React.js or ReactJS) is a free and open-source frontend JavaScript library for building user interfaces or UI components.

1.4. Reference Material

(Class (computer programming), n.d.)
(Spring Boot - Introduction, n.d.)

(Object-oriented programming, n.d.)

(C4 Model for visualising software arhitecture, n.d.)

(Why are Developers Planning to Drift Towards React Instead of Angular, 2021)

(Programming Principles, n.d.)

(What is an API? (Application Programming Interface), n.d.)

Getting Started. (n.d.). Retrieved from Axios-HTTP: https://axios-http.com/docs/intro

MULDERS, M. (2019, 09 16). *What Is Spring Boot?* Retrieved from What Is Spring Boot?: https://stackify.com/what-is-spring-boot/

Software Interfaces. (n.d.). Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Interface_(computing)#Software_interfaces

2.System Overview

For the 3rd semester of ICT & Software Engineering, students were asked to make a full-stack web application. The project must contain certain guidelines and requirement by using the agile software development methodology. More information regarding the documentation can be found in the "Documentation" folder. The web application will work as a cinema app where clients can buy tickets for movies, receive vouchers if they go quite often to the cinema as "Loyalty" system and send complaints in case something has occurred during their visit, and where admins can update the schedule, add new movies, view complaints, and send vouchers to clients. The main users of this application are client and admin.

3.System Architecture

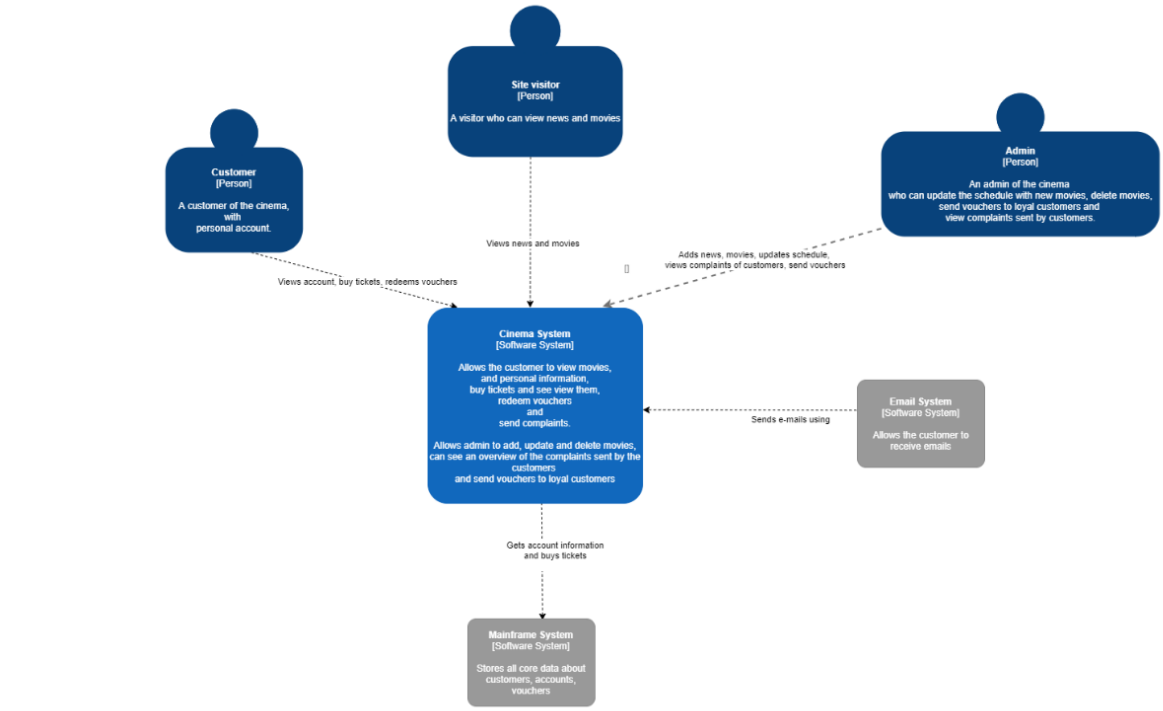
3.1. Architectural Design

The application is developed keeping in mind the SOLID principles and making sure that security is provided to the users. Interfaces are used to ensure the connection to the database in order to avoid repetition in the implementation. The backend of the application is connected with the front-end by using HTTP requests with AXIOS in order to connect to the REST endpoints. The C4 diagram below shows how the system works.

C4 diagram consists of 4 levels: System Context(C1), Container (C2), Components (C3) and Code(C4).

LEVEL 1: SYSTEM CONTEXT

System Context (C1)

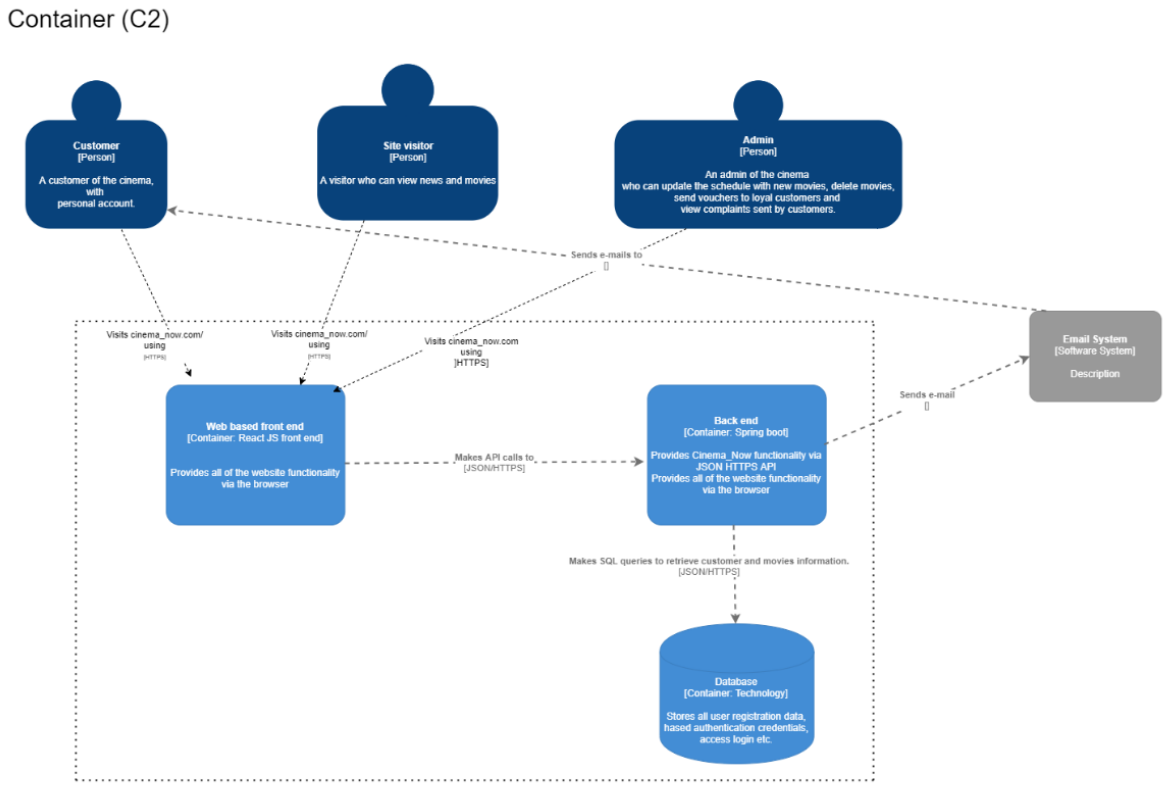


The above diagram shows the different users of the application: Site Visitor, Client and Administrator, each user having different type of accessing the website.

Cinema System is the main software part which allows the client to view movies and news, send complaints and buy tickets while the site visitor can only view news and movies. The administrator is the one responsible of managing the clients, movies, complaints, vouchers etc.

The system uses an email system so that the client can receive offers, vouchers and reset password.

LEVEL 2: Containers



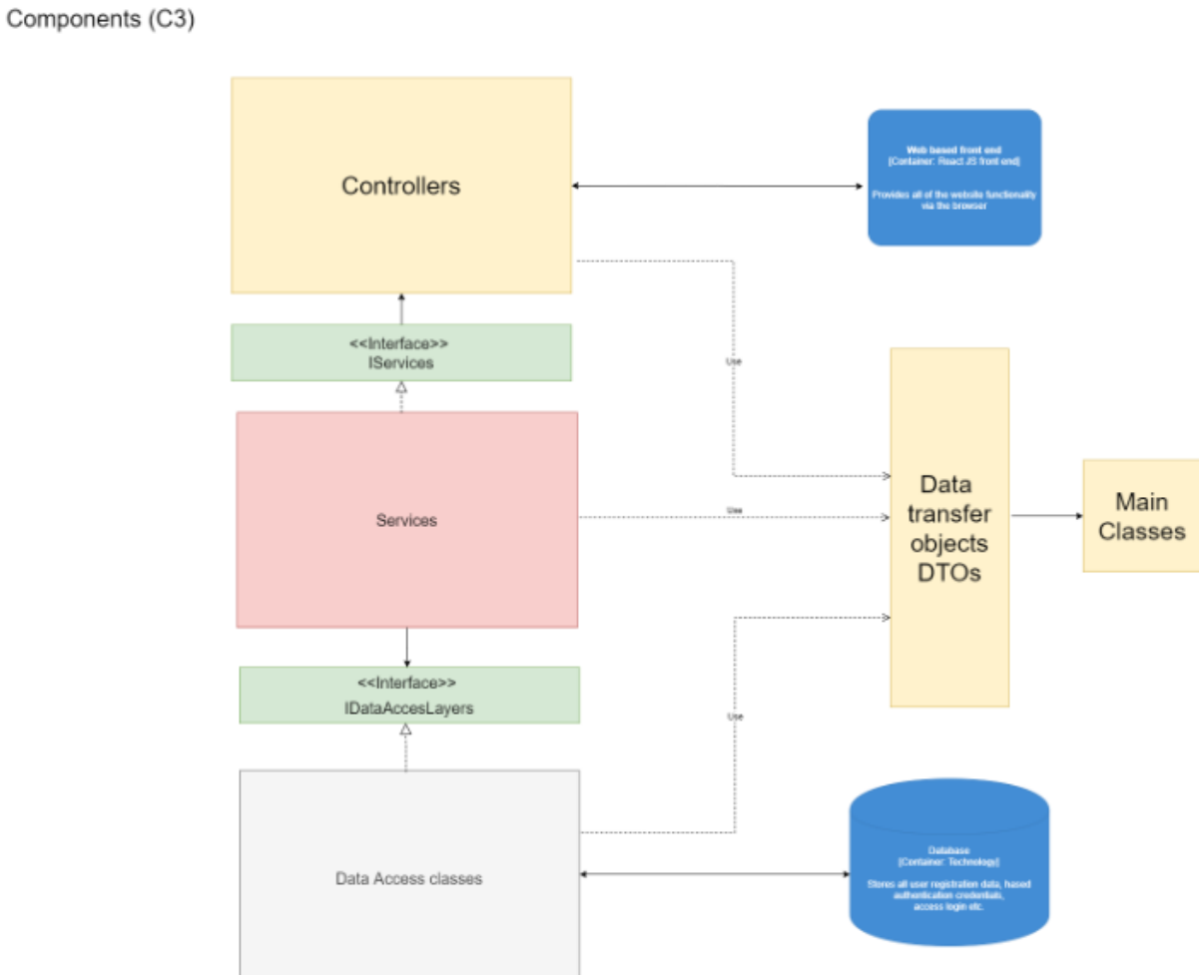
This diagram describes the connection between users and systems, and between back-end and front-end with the database.

Front-end provides the UI and all website functionality within the browser. Users and admins user the UI to actually interact with the back-end.

Back-end provides all the data and logic for the website by using Sprint Boot and JSON, therefore making the front-end functionalities possible.

Database is used to store all the data of all users/content and is directly interacting with the back-end.

LEVEL 3: Components



Level 3 represents all components which are used in the functionality of the application.

Controllers are used in order to connect the back-end with front-end. With the help of interfaces, the connection with the service components is flexible.

Service components are the logic which the controllers basically use. They are directly connected to data access layers and can CRUD data.

DTOs are used for transferring the relevant data to the front-end. We can't transfer the whole object; so, "subclasses" (DTO's) of the main ones are used to send only the relevant information.

Database is used to store all the data of all users/content and is directly interacting with the back-end.

3.2. Software choices

3.2.1 Spring Boot

According to (MULDERS, 2019) Spring Boot is an open-source micro framework, maintained by a company called Pivotal which provides Java developers a platform to get started with an auto configurable Spring application. Therefore, Spring Boot is a good framework to get started with the back-end part for full-stack web development, especially for beginners thanks to the fact that it reduces development process and increases efficiency by having a default setup for unit and integration tests. Therefore, for this semester we are obliged to use Spring Boot.

Spring Boot offers some advantages such as productivity, reduction of development time and it's easy to understand. The framework's goals are to avoid complex XML configurations (this is a big plus especially for people who are beginning full stack web development) and developing Spring applications in an easier way.

The most important thing about Spring Boot would be that everything is auto configured, so there is no need for manual configurations.

3.2.2 Separating Concerns

Separating concerns stands for splitting a computer program into distinct sections. That would mean that each component of the app has to follow a layered design (e. g. 3 presentation layer – logic layer – data access later). By using this design principle, the code is easier to understand and extend it with new features.

3.2.2 React JS

Nowadays, there are a lot of frameworks to choose from, ranging from React JS, Angular, Vue JS, Django etc. each one having its unique functionalities depending on the complexity of the project. Since, Cinema_Now is not a very complex project, it has been decided to use React JS. This framework works with components, each function being taken into account as a component. Therefore, separating concerns is crucial. For beginners, React is much more easier to get started with comparing to the other mentioned frameworks.

Also, React is quick, efficient, works with an MVC template and makes creating front-end more easy. Thanks to the fact it has a large community, you can easily find anything about different problems you might encounter while creating your application. Therefore, for this semester we are obliged to use React JS.