Design Document Individual Track Bamboo Restaurant

Eindhoven, October 5th, 2021

Table of Contents

Revision History:

Date Changed	Document Version	What was changed
05/10/2021	1	
26/11/2021	2	Update the C4 diagrams
14/12/2021	3	Update the C4 diagrams

The Choice of Technologies

For this project, technologies will be used are Java and Spring Boot for the backend server side, HTML/CSS/JS and React for Frontend web application, MySQL will be the main database techology.

Java is the OOP programming language, and has been the most popular programming language choice by developers and companies for a long term of period. Java has strong features of security and performance.

Spring Boot is a Java framework which mainly used for Server Side applications. Spring Boot provided a lot of library for security and performance to reduce the implementation needed and have been used by companies.

React is the Javascript library that supported by Facebook and its community. React provides easy implementation compares to Vanilla Javascript.

Seperating the concerns in React

My strategy is using 3 tier design, separate the api connections, the logic to process the data from apis, and the UI code for React itself. So that the parts to handle processing the data do not to know the detail of connection apis they receive from, and the React UI will receive the data from the logic processing and rendering it.

```
async function GetFromDatabase(){
    let response = await axios.get("http://localhost:8080/reservation/me");
    let data = response.data;

    const listOfReservationCards = data.map((detail) =>
        <ReservationCard date={detail.date} startTime={detail.startTime}
                         numberPeople={detail.numberPeople} description={detail.description} />
    );
    console.log(listOfReservationCards);
    return (listOfReservationCards);
}

class ReservationPage extends React.Component{
    constructor(props) {
        super(props);
        this.state = {content: null};
    }

    async componentDidMount() {
        let newContent = await GetFromDatabase();
        this.setState({content : newContent});
    }

    componentWillUnmount() {
    }

    render(){
        return (
            <>
            <h1>My History of Reservation</h1>
            {this.state.content}
            </>
        );
    }
}
```

Figure 1. Separating the concerns in React

The C4 Diagrams

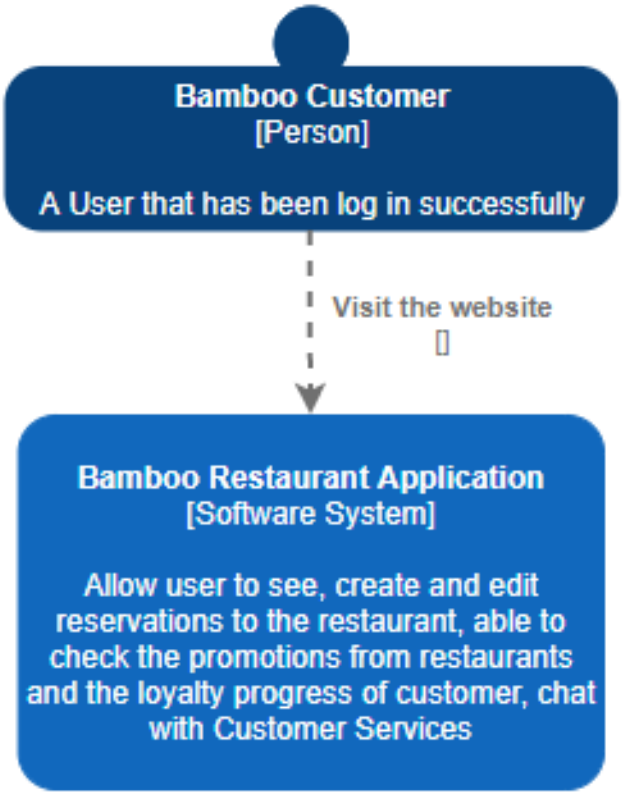

Figure 2. Reservation Context Diagram

Firstly, customers if successfully login to the Bamboo Restaurant, they will see the dashboard for reserving a table in the restaurant, which shows the history of reservations have been made and the form to book new reservation. The dashboard connect to the backend server, which stores and processes the booking table from the customers.

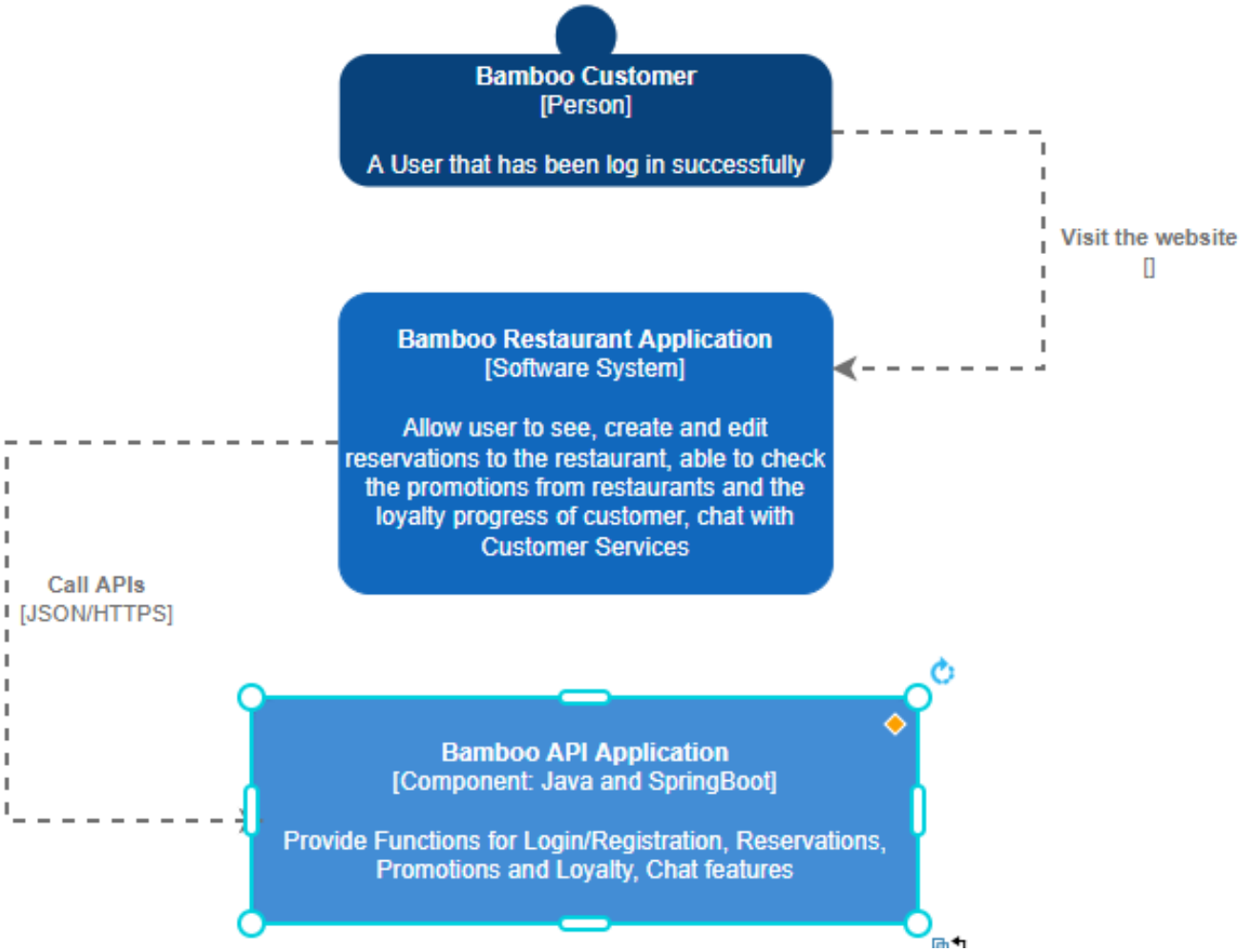

Figure 3. Bamboo Container Diagram

Website will be based on React framework, as the website will be made from components that we can reuse again for fast development, also React support collection of libraries recommended by developers commnuity, help creating applications more effcient without to focus on the security and performance – libraries will help. The website will communicate to Reservation apis by using JSON/HTTPS protocol, as it is secure for transmitting the data.

The reservation apis will be built on Spring Boot framework, which also supports libraries for fast developing, Spring Boot Standard Libraries supports authenticate and authorization, database driver to connect our SQL Database.

Our database use MySQL engine on the development phase, may be in the future, we may use different technologies if the performance is the criteria from our clients.

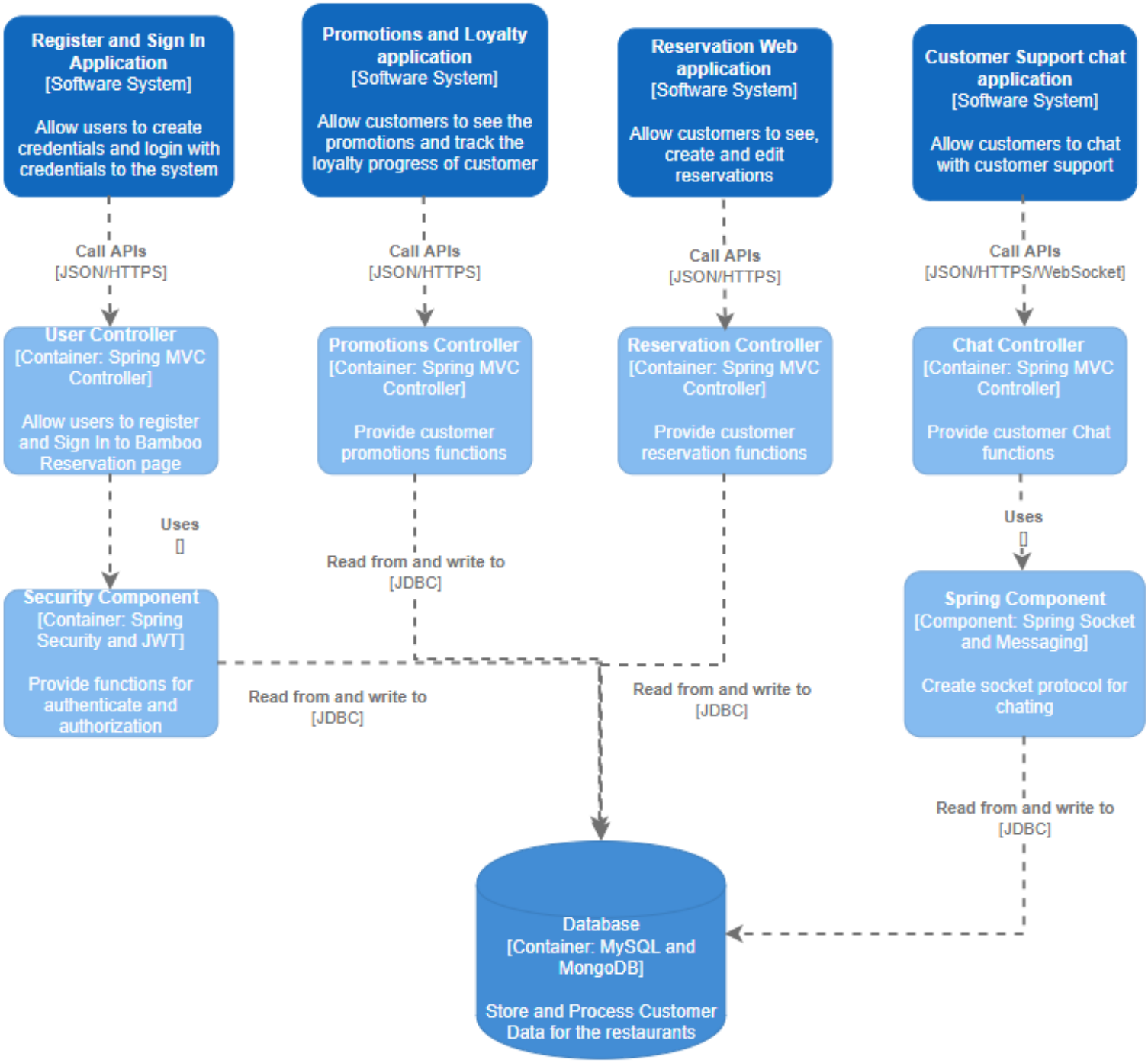

Figure 4. Bamboo Components Diagram

Our Reservation apis will have Reservation Controller and Sign In Controller for authorizing the customers access the Reservation Controller. Sign In Controller will use the Security Component build on the Spring Security and JWT (Java Web Token), supports features for the security without need to build on scratch again. Our customers credentials will be stored in the MySQL and it may change in the future if clients want a high performance application after building the features.

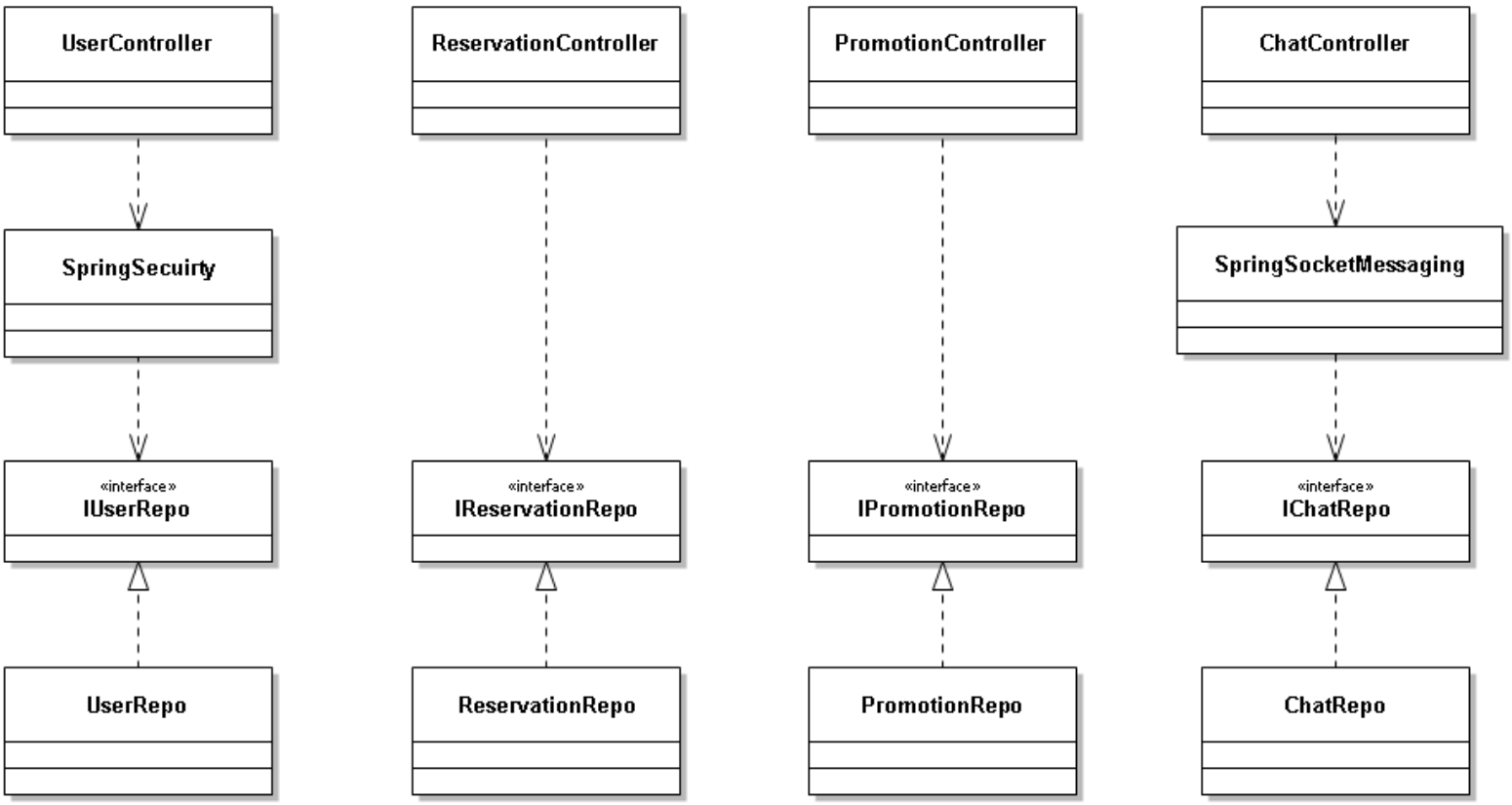

Figure 5. Bamboo Code Diagram

As from the diagrams, LoginController will use Spring Security as it is one of the standard library provided from Spring Boot, however our UserRepo using MySQL that not provide from Spring Boot, so we have to apply interfaces to UserRepo as our SpringSecurity may use different database engines.

The UML Diagrams

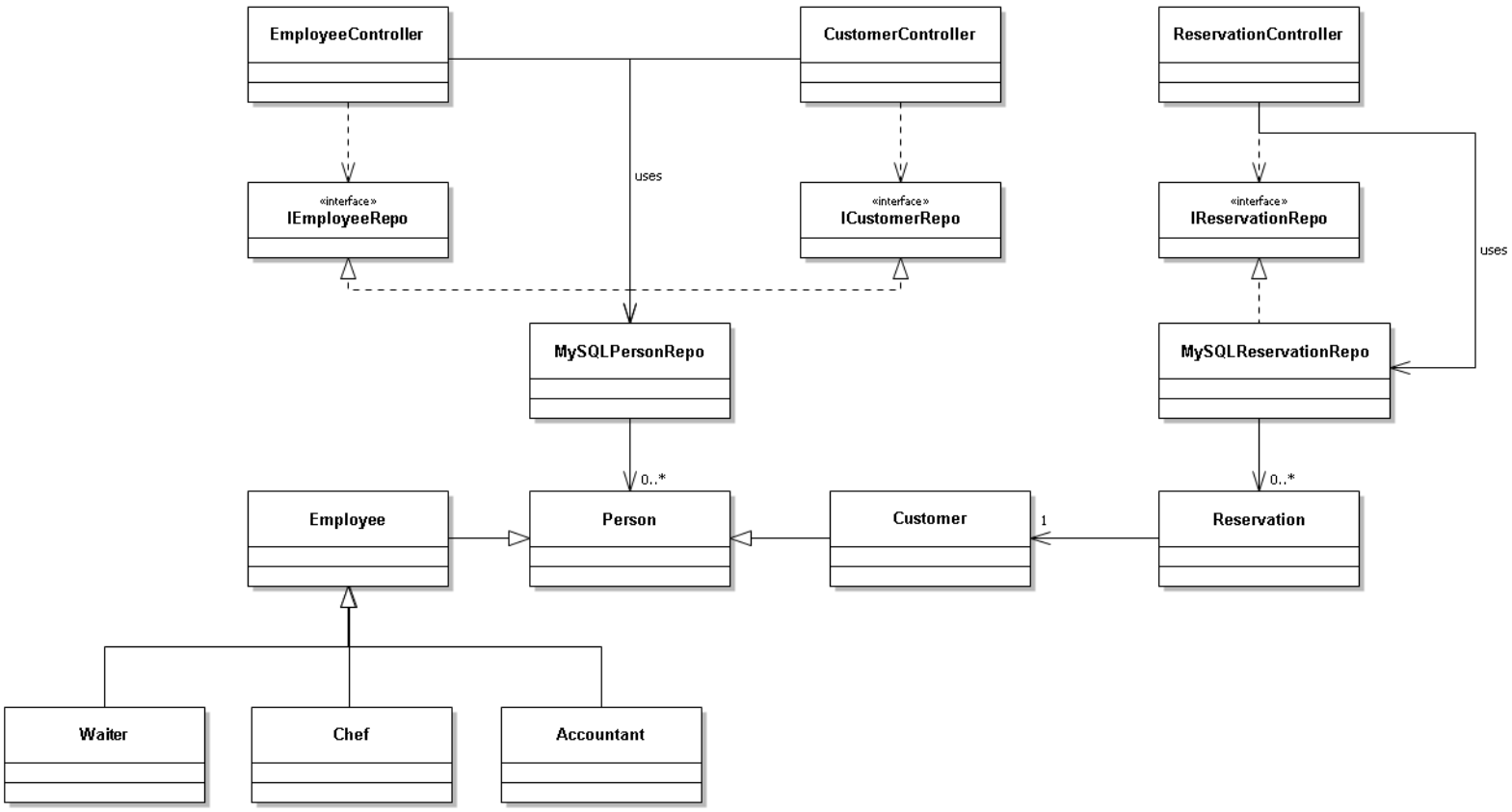

Figure 6. The UML Reservation Diagram

High Level Design and SOLID Principles

From the UML diagram above, the design consists of 3 controllers to handle the authenticate, authorization and the reservation system. Foreach controller, it will need the data access layer (Repo – Repository). My implementation will use interfaces, as D in Dependency Inversions, our components will depend on abstraction not on the concrete implementation, and benefits from using the interfaces that I can extend classes from chosen interfaces and injecting to a higher classes without changing the details of current implementation, like hot swapping .