

This report

describe

s

the process which will be

used for creating the Crew Logistic System

Project

Report

Interim

Submission

**CS6P05 Project**

Name:

Arkadiusz Grudzien

ID Number:

20001306

Date:

Sunday, 08 January 2023

Module leader

:

**Dr.**

**El**

**a**

Homayounval

a

First

Supervisor:

Victor Sowinski

-

Mydlarz

Secon

d Supervisor

:

Sandra Fernando

# Declaration

**PLAGIARISM**

You are reminded that there exist regulations concerning plagiarism. Extracts from these regulations are printed below. Please sign below to say that you have read and understand these extracts:

Student signature: Arkadiusz Grudzien Date: 08/01/23

This header sheet should be attached to the work you submit. No work will be accepted without it.

|  |
| --- |
| Extracts from University *Regulations* on Cheating, Plagiarism and Collusion  Section 2.3: "The following broad types of offence can be identified and are provided as indicative examples…   1. Cheating: including taking unauthorised material into an examination; consulting unauthorised material outside the examination hall during the examination; obtaining an unseen examination paper in advance of the examination; copying from another examinee; using an unauthorised calculator during the examination or storing unauthorised material in the memory of a programmable calculator which is taken into the examination; copying coursework. 2. Falsifying data in experimental results. 3. Personation, where a substitute takes an examination or test on behalf of the candidate. Both candidate and substitute may be guilty of an offence under these Regulations. 4. Bribery or attempted bribery of a person thought to have some influence on the candidate's assessment. 5. Collusion to present joint work as the work solely of one individual. 6. Plagiarism, where the work or ideas of another are presented as the candidate's own. 7. Other conduct calculated to secure an advantage on assessment. 8. Assisting in any of the above.   Some notes on what this means for students:   1. Copying another student's work is an offence, whether from a copy on paper or from a computer file, and in whatever form the intellectual property being copied takes, including text, mathematical notation and computer programs. 2. Taking extracts from published sources *without attribution* is an offence. To quote ideas, sometimes using extracts, is generally to be encouraged. Quoting ideas is achieved by stating an author's argument and attributing it, perhaps by quoting, immediately in the text, his or her name and year of publication,   2  e.g. "e = mc (Einstein 1905)". A *references* section at the end of your work should then list all such references in alphabetical order of authors' surnames. (There are variations on this referencing system which your tutors may prefer you to use.) If you wish to quote a paragraph or so from published work then indent the quotation on both left and right margins, using an italic font where practicable, and introduce the quotation with an attribution. |

# Abstract

This project requires the development of a web-application system for the event company Pinnacle Crew. Pinnacle have been specializing in providing crew for corporate events. They have systems and tools in place for maintaining their business but are missing valuable features meaning they don’t currently have a system that can efficiently maintain organisation of the employed crew within their company. The system currently in use is a simple SMS system, which is not convenient from the crew's perspective. The current system functions in the following way: The admin (officer) sends an

SMS to each crew member detailing: Job Number, Date, Time, Required Amount of Crew, Shift Duration, Client’s Company Name, Contact Number on site, Additional Notes. When the crew receive the message, the user must reply with 'YES' or 'NO' and supply the job number ID. The confirmation, if the crew member is assigned to the job, is called 'CREW LIST'. This is sent usually 24h before starting the job. The crew list consists of all the information about the job, phone numbers and names of all the members which they have been assigned to the specific job. The idea for the new system is to substitute the current inconvenient system to web-application where crew members can effectively communicate with the office about the jobs. The employers also will see the details about the jobs, their details etc.

# 1: Introduction

## 1.1 Project topic and rationale

The project topic is a web application called Crew Logistic System. The functionality is like already existing system such as Employee Management System. The motivation which has been pushed me to create the project is that I am part-time employee of the company. During work at the company, I have realized what are the problems and how current system is inconvenient from the crew’s perspective. I have decided to develop a web application to solve the issues.

The software will be designed to help the company Pinnacle Crew to manage their crew members and jobs. The project is interesting because it is a real-life project, and it is a good opportunity to learn how to develop a web application. The problem in hands is that the company Pinnacle Crew is using a SMS system to manage crew member. The jobs are events that the companies of events are organizing, and the crew members are the people that are working on the events. The admin sends jobs to specific crew members. The crew members must confirm or reject by doing specific action in the app. The usefulness of the app is that it will help the company to manage their crew members and jobs more efficiently.

## 1.2 Projects Aims and Objectives

1.2.1 Aims:

Design and implement a full system web application which enable to change the communication between the office (administrator) and the crew. There is going to be fixed couple things which they are inconvenient in the current system. The most important thing which is going to be fixed is that the crew have a problem with tracking their jobs and hours. The system will count the hours and store all jobs which has been done by each worker (crew). The system will change the work how administration office works. The office will not send messages anymore to the crew with jobs, all these will happen in the web application. They will create a job and upload to the system to be seen by the crew. The crew will see available jobs received from admin and will accept it or decline. The system will have an account system for each individual admin and crew. The system will store their details such as documents, avatars photos etc. The system will be designed using new technologies, such as Spring Boot as a backend system. All the logic of the system will be written there and executed on the server where the application will be deployed.

1.2.2 Objectives:

Write a full backend system in Spring Boot including following functionalities:

Account where user will be login in. The user will login as an admin will see the dashboard as a landing page. The dashboard will consist of all the brief information about the activities in the system such as:

* Who recently joined to the company.
* Recent added jobs (it must be because there will be more than one admin (officer)) o Ability to redirect to those sections in the system from dashboard page.

➢ The admin also will have following sections:

User management – admin can add another user to the system choosing role as admin or crew member. Also, admin will have the ability to edit details, delete user and suspend account.

Jobs management – admin can send a particular job to chosen crew members so only crew members with required skill wills receive a job where the skills are necessary.

Profile – despite who is logged in (admin or crew member) will have the ability to change the details, upload documents, change avatar etc.

Logout – for invalidate a session for logged user.

The system also will have a feature where the user will receive an email that his/her account has been created. The user must click on the link in the email to activate their account.

Make a design for frontend in Figma and then create it in React JS.

## 1.3 Methodology

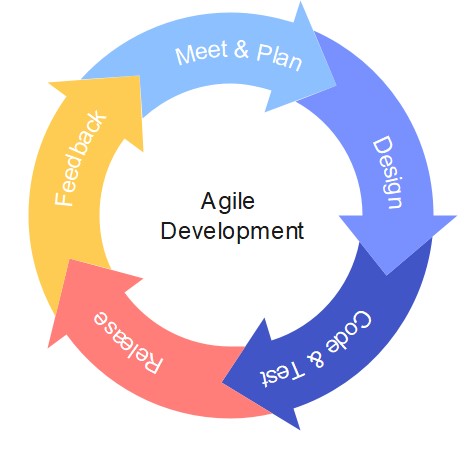
The methodology which I am going to use for this project is called Agile methodology. Agile is a method for creating a software. It’s a chain between rapid development and deployment.

This process emphasizes flexibility and rapid iteration. It is based on the principles outlined in the Agile Manifesto, which values individuals and interactions, working solutions, and customer collaboration over processes and tools.

Agile development is characterized by short development cycles, called "sprints," which typically last a few weeks. During each sprint, developer works on a specific set of features or tasks. The creator or team uses regular meetings, called "scrum" meetings, to communicate progress and address any issues that may arise.

One of the key principles of agile development is the concept of "iterative development," in which the creator continually refines and improves the product through multiple cycles of development (Figure 1). This allows the developer to quickly incorporate feedback from users and stakeholders, and to adapt to changing requirements or priorities.

Overall, the goal of agile development is to deliver high-quality products to customers in a timely and efficient manner, while also fostering a culture of continuous learning and improvement.



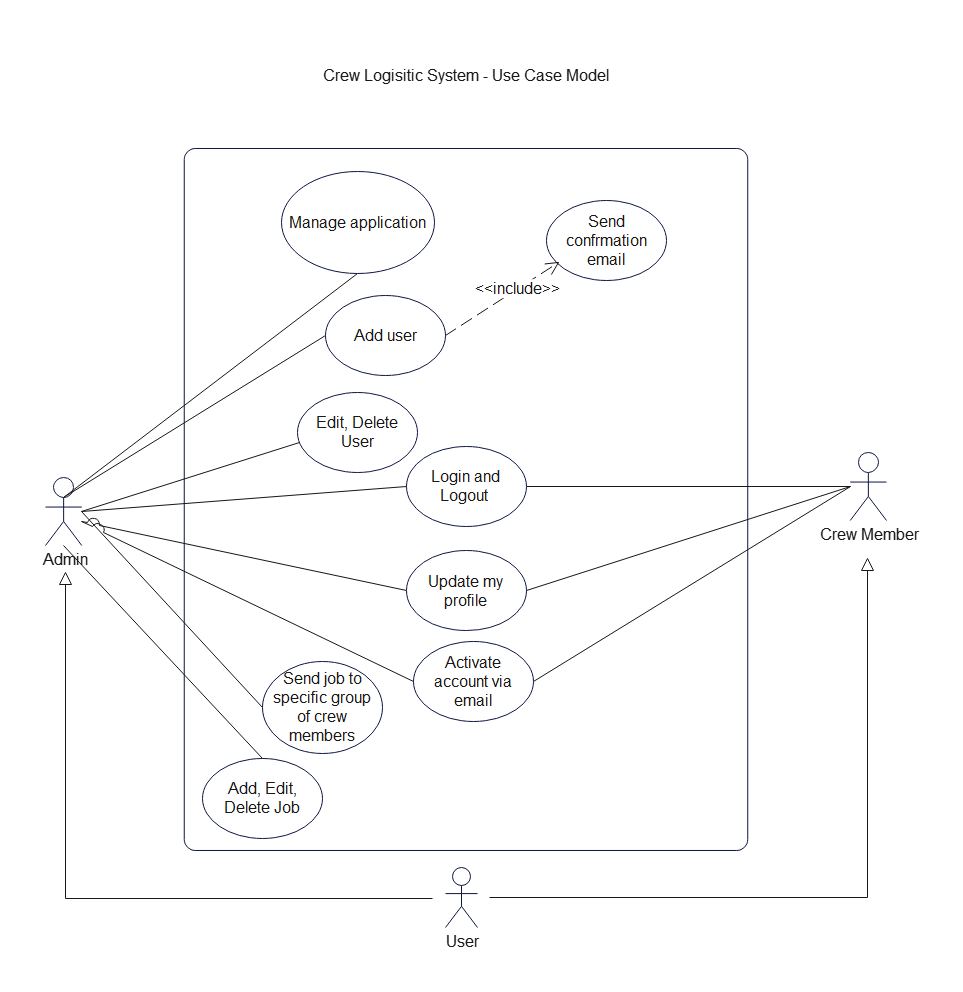
*Figure*

*1*

*Agile Development Process*

1.3.1 Use Case Model

Use Case model performs the functions which will be available in the system. During the process of developing the function might be reduced or expanded.



*Figure 2 Crew Logistic System - Use Case Model*

1.3.2 High Level Use Case Descriptions

**Use case:**  Manage application

### Actor: Admin

**Description:**  The admin who is logged in is responsible for maintain the application and check if every function is working correctly, also needs to check the logs to be able to see if there are some issues with backend. If some errors occurred needs to inform immedia tely the team who made the application.

**Use case:** Add new user

### Actor: Admin

**Description:** The admin login into his account and then go to section where new user can be created. Needs to fill a form with details about the new user. Ones form is complete; the admin clicks on a button “create new user”. Ones the button is clicked the system sends notification to the email for the new user.

**Use case:** Edit, Delete User

### Actor: Admin

**Description:** Ones the admin is on his/her account can modify the details about chosen user it can be admin or crew member. It depends on what is needed. The admin clicks on appropriate button edit or delete user. If admin clicked on edit button, then appears a form with data loaded from database. At this point user can

edit the details about user. There is also a button for completely deleting the chosen user.

### Use case: Login

**Actor:** Admin, Crew Member

**Description:** The user can login through the main page of the application by providing correct details about the account – email and password. Ones all data are correct the page is redirected to dashboard for the login user. It’s different accordingly for admin and crew member. The system saves JWT – Jason Web Token in the browser. The JWT is a string which contains all the details about logged user. The JWT work as a session for the user.

### Use case: Logout

**Actor:** Admin, Crew Member

**Description:** The user can logout by clicking on appropriate button which is positioned on the navigation bar. Ones the user is logged out the JWT is automatically deleted.

**Use case:** Update my profile

**Actor:** Admin, Crew Member

**Description:** The user can update their profile by clicking on the button available on the sidebar. The users can change the details about themselves also can upload documents such as passport etc. It will be decided in future development what kind of documents are needed.

**Use case:**  Activate account via email

**Actor:**  Admin, Crew Member

**Description:**  Ones the user is created needs to click on the received link. When the user does that is redirected to the page where can login to the system.

### Use case: Send job Actor: Admin

**Description:** Admin can create a group from a list of available crew and then send a specific job to those crew. This is useful because if any crew who doesn’t have required skill will be skipped.

**Use case:** Add, edit, delete job

### Actor: Admin

**Description:** Admin can create a job. The job can be edited, changing the details about the job. The job can be also removed by admin.

# Chapter 2 Background Research

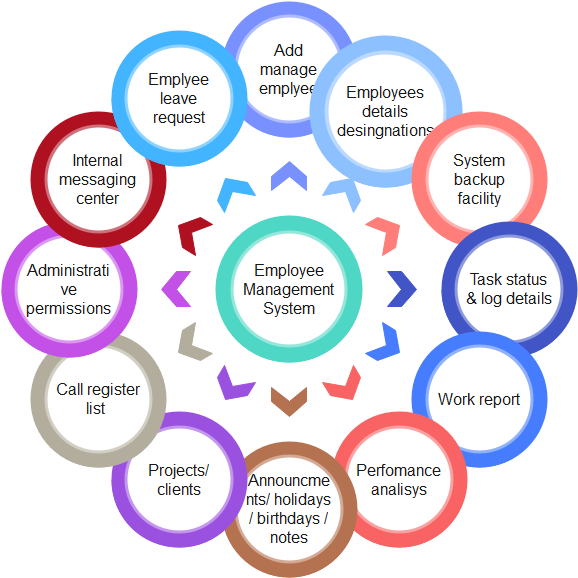
List of similar projects which will help to make this project:

* EMS – Employee Management System
* MSS – Management Self-Service
* HRIS – Human Resource Information System
* HR – Human Resource
* HOD – Head of Department

2.1 Literature reviews on subjects related to the project.

**EMS**

The employee management system gives efficient way to manage work between admin work and the team. The system is designed to help the business owner specially to save money and time. The system can help for example what kind of task does specific worker. The admin via the system can assign to different assignment or/and department. The worker can efficiently apply for a holiday and the admin can accept it or reject it. The employees can add work report. Employee also can send a leave request so the employer will know this information immediately and will start to look for a substitution.

The Employee Management System is very important for the business where it runs. The data which is store in database can be accessed just for the authorized personnel with specific permissions. This is one of many important aspects during developing such as system. The system allows to have the data from the origin source which means from the employee because he/she can update the details about their self. The system provides a quick information for the admin and especially for the worker. The worker can see the work schedule, salary information, education information or contact information. This data is very useful because it allows to improve the efficiency and workflow of the business. Based on these data the admin or head of the organization can assign or reassign the specific tasks or departments or adjust the salary and the work schedule etc. [1]

*Figure 3 Example of functions for Employee Management System*

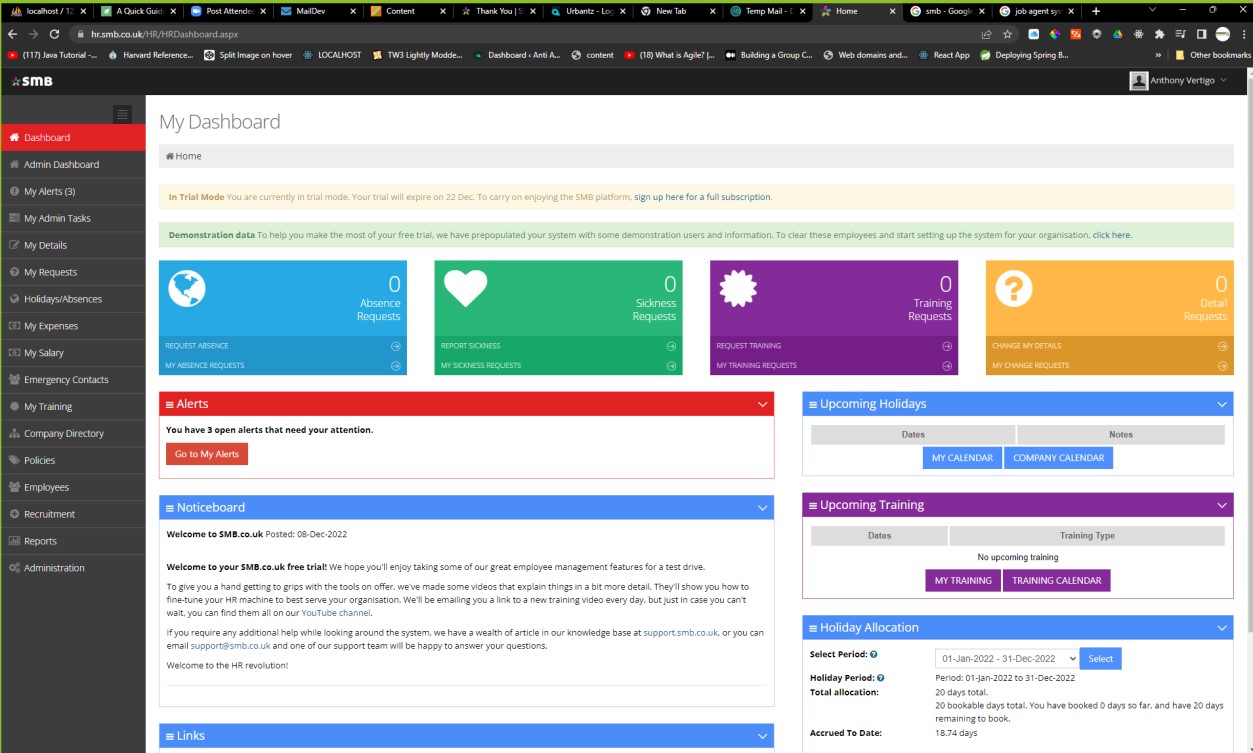
**HRMS**

Human resource management system refers to the systems and processes that interface between human resources Management (Human Resource Management) and Information Technology. It regards human resource management as a discipline with Basic HR activities and processes, especially in the field of information technology The programming of data processing systems evolves into standardized routines and packages ERP (Enterprise Resource Planning) software. [2]

## 2.2 Evaluation of related products/solutions

**Employee Management System example 1**

I have listed below few similar products which they are available online. **smb.co.uk**

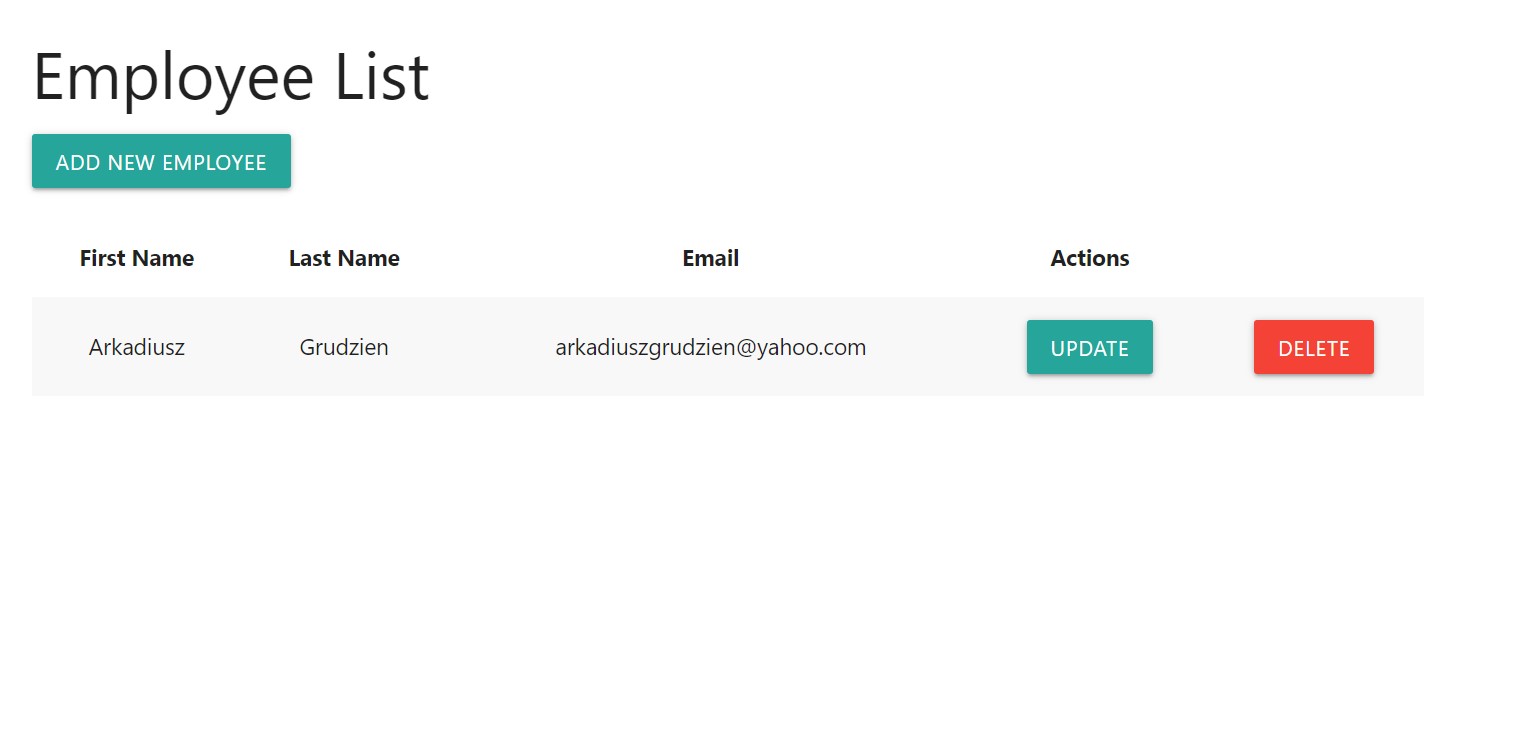


*Figure 4 Dashboard and Navbar of SMB*

SMB helps for employers to manage data about their employees. It has friendly user interface and intuitive user experience. I will use it to inspire and adapt some features and solution from this platform.

**Employee Management System example 2**

The figure below preform the basic Employee Management System found in GitHub repository.



The project is a basic EMS system where user can add an employee. Ones the button “Add new employee” is clicked the page is redirected to form. The form contains fields about new employee who can be added by clicking “Add employee”. The employee can be also edited and deleted form the database. The system using MySQL server. This project allows me to have a basic concept for my application. This project has been created in Java 11 and Spring Boot. I can see what functions has been created and see what I can use to my project. The fronted has been created in Thymeleaf the template engine for Spring Boot. In my case therefore will be created in React which is more convenient end efficient.

## 2.3 Scope of the project

Goals:

Login system where admin or crew member will be logging in

Dashboard where admin can see what has been done in the system such as: new crew is available to work:

Adding new jobs and users.

Deleting, editing jobs and users.

Profile section where user can edit details, add documents etc.

## 2.4 Tools/solutions selected to use in the project

2.4.1 Spring Boot – backend

Spring Boot is a Java-based framework used to build standalone, production-grade applications. It is designed to simplify the process of developing and deploying applications by providing a set of tools and libraries that make it easy to create and run a new application.

Some of the key features of Spring Boot include:

* An embedded servlet container (such as Tomcat or Jetty) that makes it easy to run the application as a standalone Java application.
* Auto-configuration of Spring beans based on the presence and configuration of certain classes in the classpath. This means that you can create a simple Spring application without writing any configuration code.
* Support for various types of database and messaging technologies out of the box, including JDBC, JPA, Hibernate, and JMS.
* A command-line tool that makes it easy to create and run Spring Boot applications.
* Support for easy deployment to various cloud platforms, such as AWS and Heroku.

Spring Boot is a powerful tool for building Java-based applications that are easy to develop and deploy.

2.4.2 React – frontend

React is a JavaScript library for building user interfaces. It was developed by Facebook and is often used for building single-page applications (SPAs) and mobile applications.

One of the key features of React is that it allows developers to create reusable components that can be easily combined to create complex user interfaces. This makes it easy to build large, interactive applications that can be easily maintained and scaled over time.

React is designed to be fast and efficient, with a virtual DOM (Document Object Model) that allows it to update only the parts of the page that have changed, rather than redrawing the entire page. This makes it well-suited for applications that need to handle large amounts of data and update the user interface in real-time.

In addition to its core library, React also has a number of complementary tools and libraries, such as React Router for managing application routing, and Redux for managing application state.

2.4.3 MySQL

MySQL is a popular open-source relational database management system (RDBMS). It is widely used for managing data in web and mobile applications, and is known for its reliability, flexibility, and ease of use.

MySQL is based on the Structured Query Language (SQL), which is a standard language for interacting with relational databases. It allows users to create, modify, and query databases, as well as control access to the data stored in them.

MySQL is designed to be fast and scalable, with a wide range of features that make it well-suited for a variety of applications. These features include support for transactions, foreign keys, views, and stored procedures, as well as support for various data types and indexes.

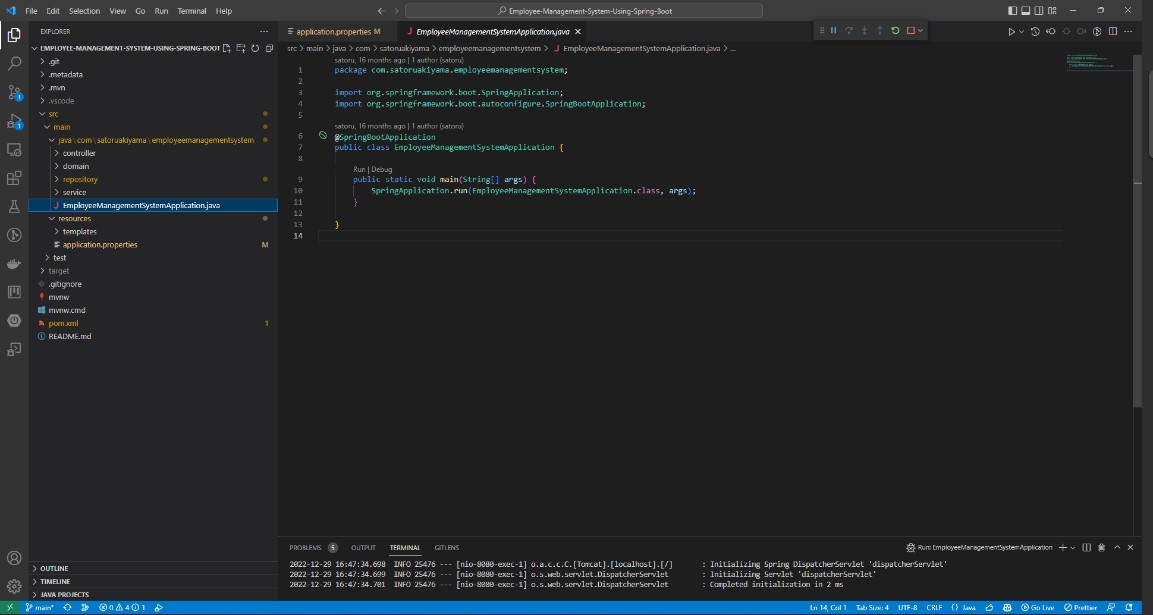
In addition to its core database engine, MySQL also includes a number of tools and utilities for managing and administering databases, such as the MySQL Workbench graphical interface and the mysqldump utility for backing up and restoring databases.

MySQL easily integrate with Spring Boot which makes the database convenient to use for this project. The platform can be installed additionally with phpMyAdmin. It’s a user interface for MySQL where can be done the CRUD operations.

2.4.4 Visual Studio Code

The program is a free open source code editor developed by Microsoft. This Integrated Development Environment is useful because there is variety of extensions to add for it. The most important are :

o IntelliSense – it help to write the code, its suggesting a code while typing. o Debugging support which allows to go through code and see potential issues. o Included terminal which allows to run a command straight away in the IDE.



*Figure 5 Visual Studio Code in action. It has intuitive user experience which helps to make projects.*

# Chapter 3 Requirements Analysis

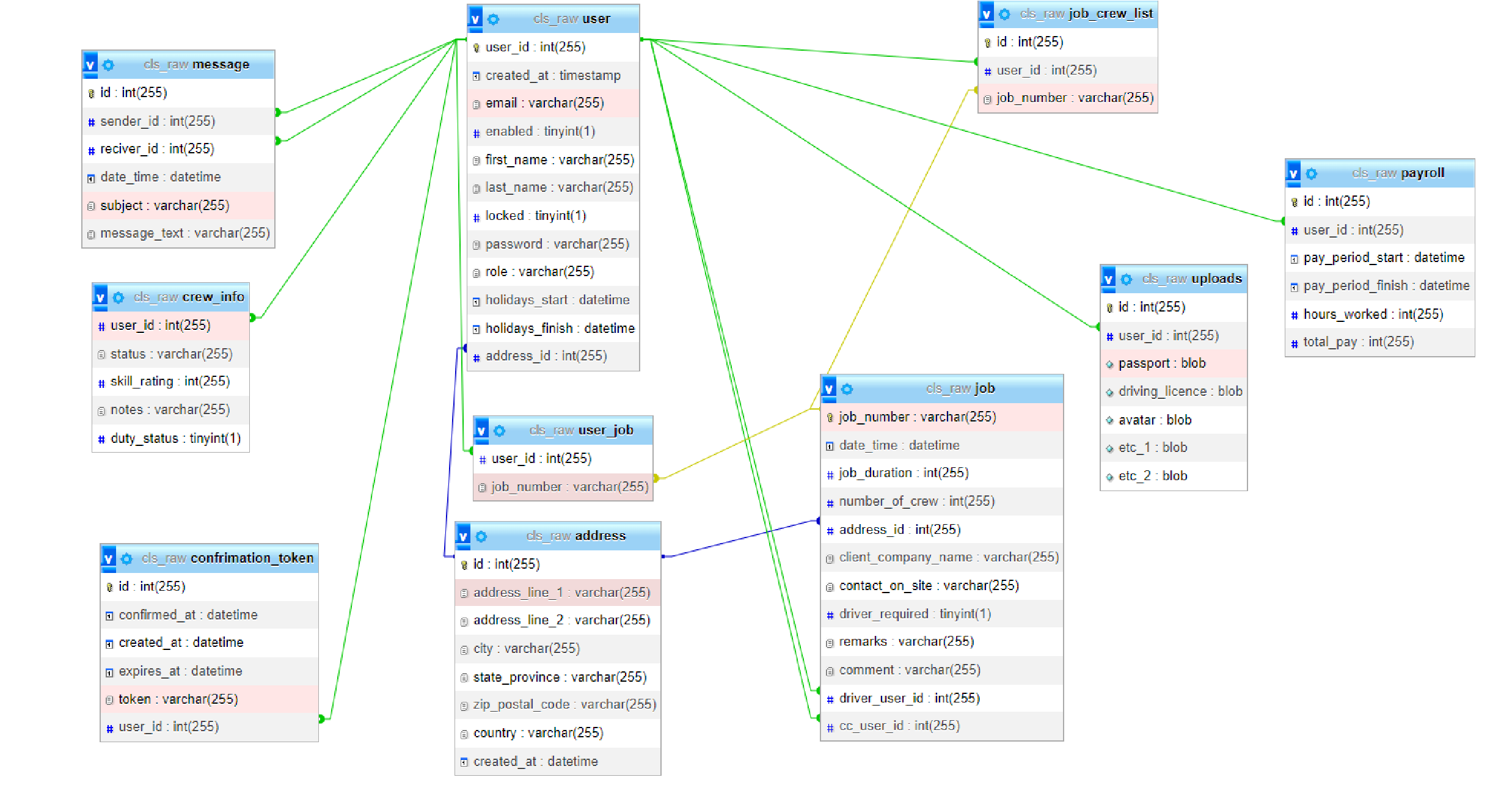
A crew logistics system is a software application that is used to manage the scheduling, logistics, and support activities related to the crew of the organization called Pinnacle Crew. The main requirements are:

* Scheduling and rostering: The system should allow users to schedule and roster crew members and jobs, considering factors such as qualifications, availability, etc.
* Crew tracking and location: The system should track the location and status of crew members, including whether they are on duty or off duty.
* Crew communication and messaging: The system should provide a means for crew members to communicate with admin, such as dispatch or ground crew, and should allow for the sending and receiving of messages and alerts.
* Training and certification: The system should track and manage the training and certification of crew members, including the expiration of licenses and other qualifications.
* Reporting and analytics: The system should provide various reports and analytics to help users understand the crew logistics data and make informed decisions.
* Integration with other systems: The system should integrate with other systems that are used by the organization, such as payroll, HR, and maintenance systems.

**Chapter 4 Software Design.**

## 4.1 Database

The database for the web application has been designed in phpMyAdmin. It’s a user interface for MySQL Database. The program has a database designer integrated in it so it’s easy to maintain and edit the database. In this case that the program will be written in Spring Boot, at the first instance I have designed test database. The schema will be generated in Spring Boot, but I have created test database to have a structure and base point to start write the program. For testing purposes, the database has been named the as cls\_raw.

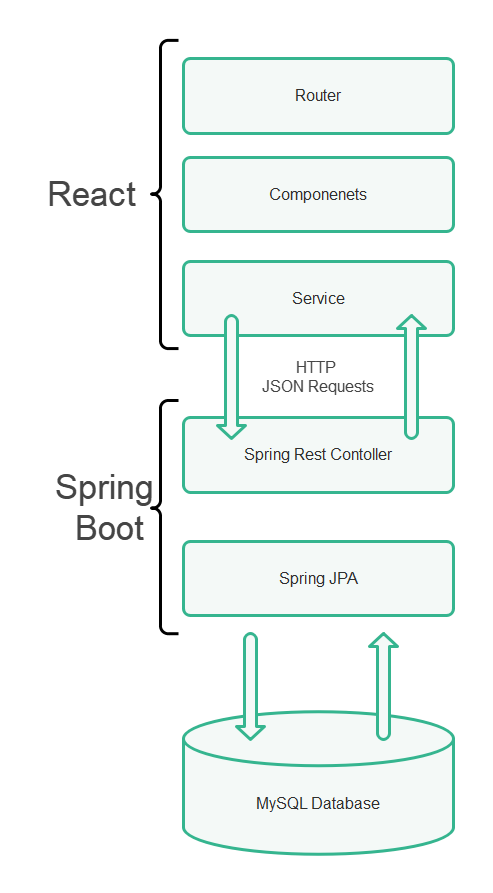


*Figure 6 Entity Relationship Diagram of the database*

This database is a breakpoint for creating the program. During development in Spring Boot the database might be a bit different from the original because Spring Boot sometimes creating additional tables for better performance.

## 4.2 Basic Software Logic

The figure below performs conceptual logic of the application. The react is responsible for handle the views. Its communicating with Spring Boot (backend) via HTTP Requests which are written in JSON (Java Script Object Notations). What it means it sends objects as JSON and then Spring Boot processing by functions written in Java. The Spring JPA is responsible for communicating between Spring Boot and database which is MySQL.

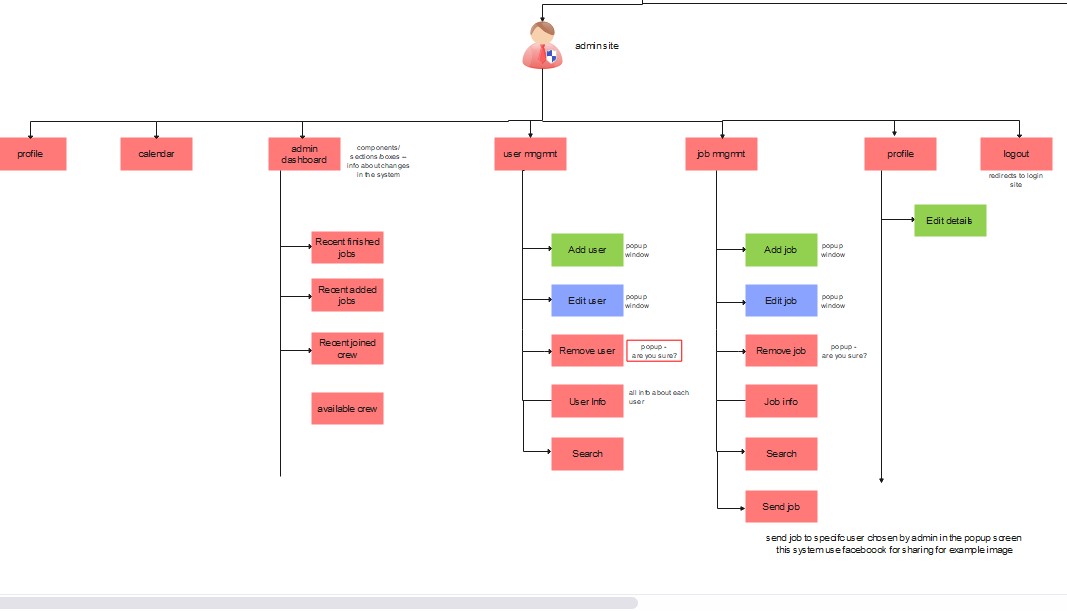


*Figure 7 Conceptual Logic Diagram of the Software*

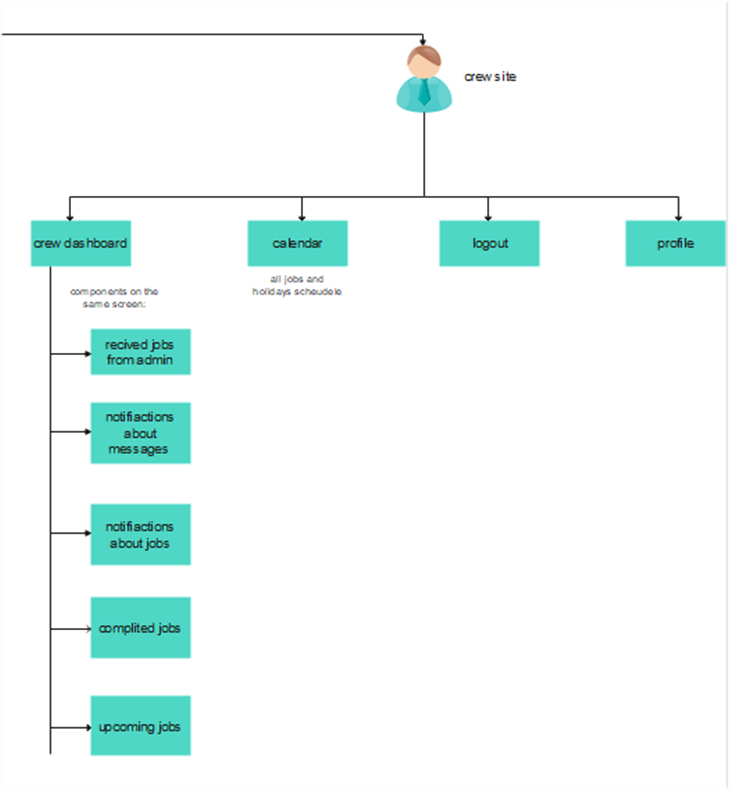
## 4.3 User Interface

User interface website structure

*Admin Site*



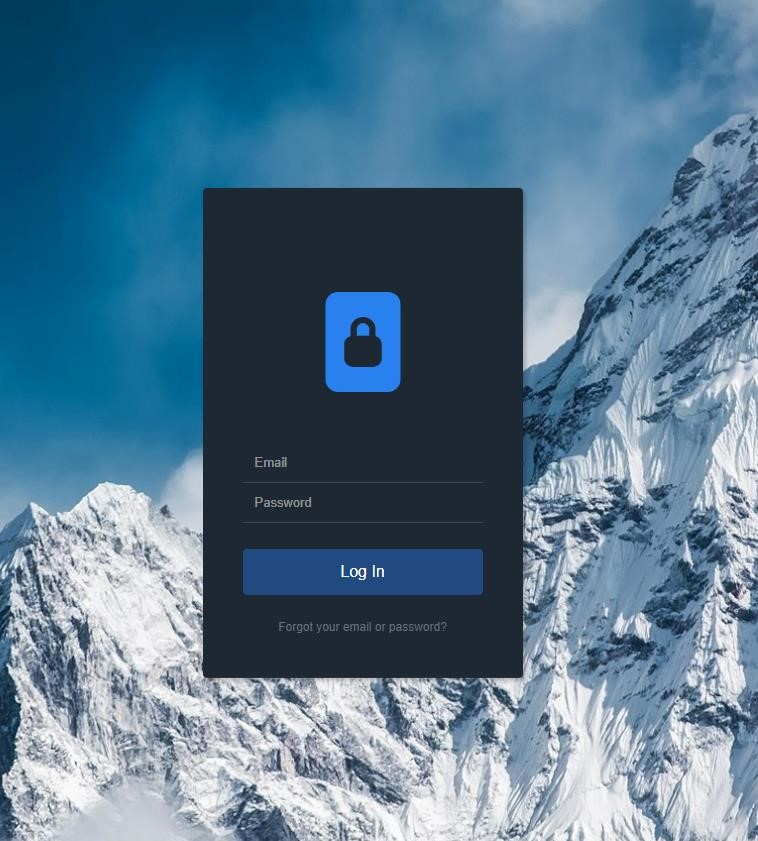
*Figure 8 Application structure - admin site*

*Crew Site*

*Figure 9 Application structure – crew site*

Login Page

Login page has been created with modifying already existing template



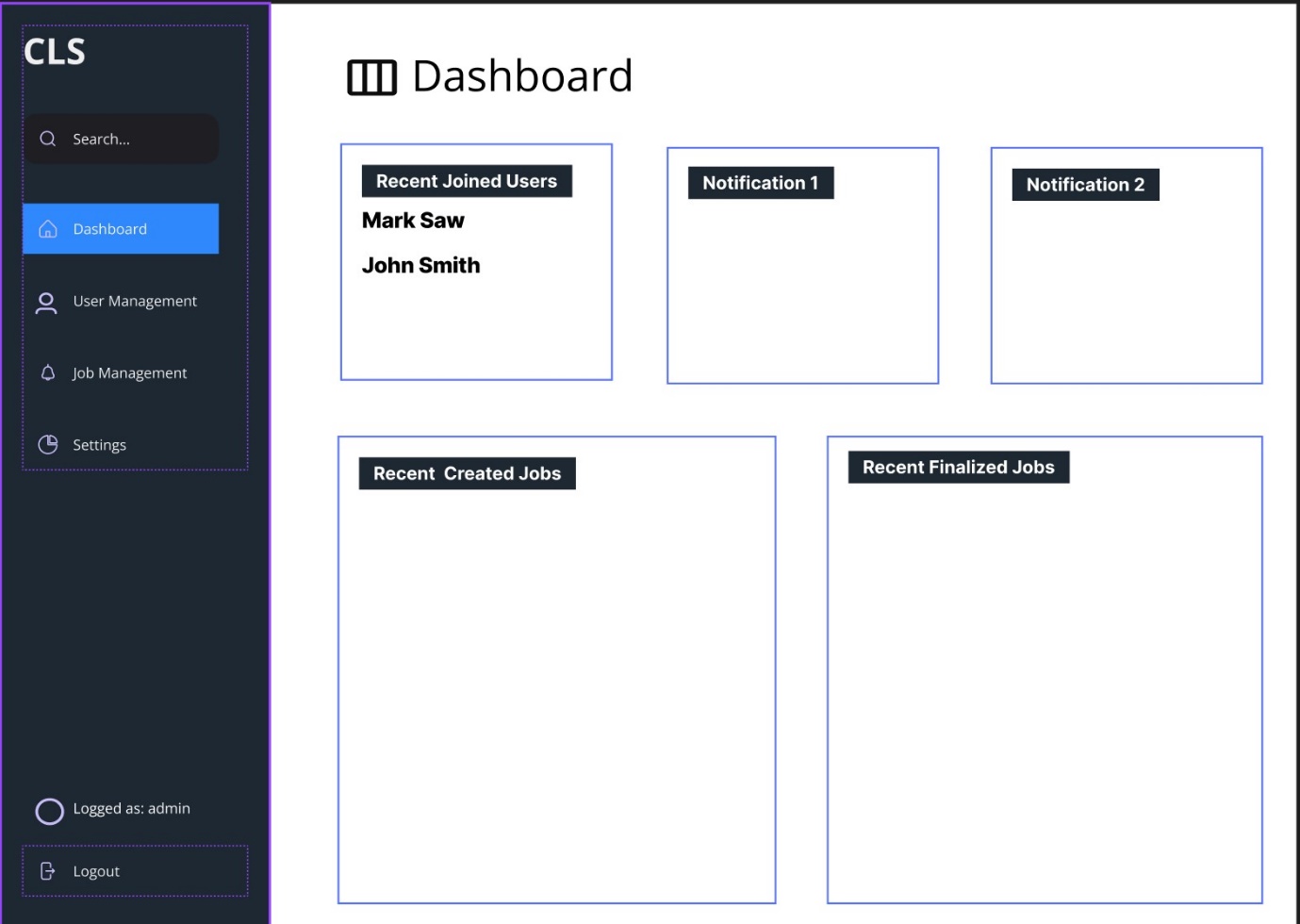
*Figure 10 Login Page*

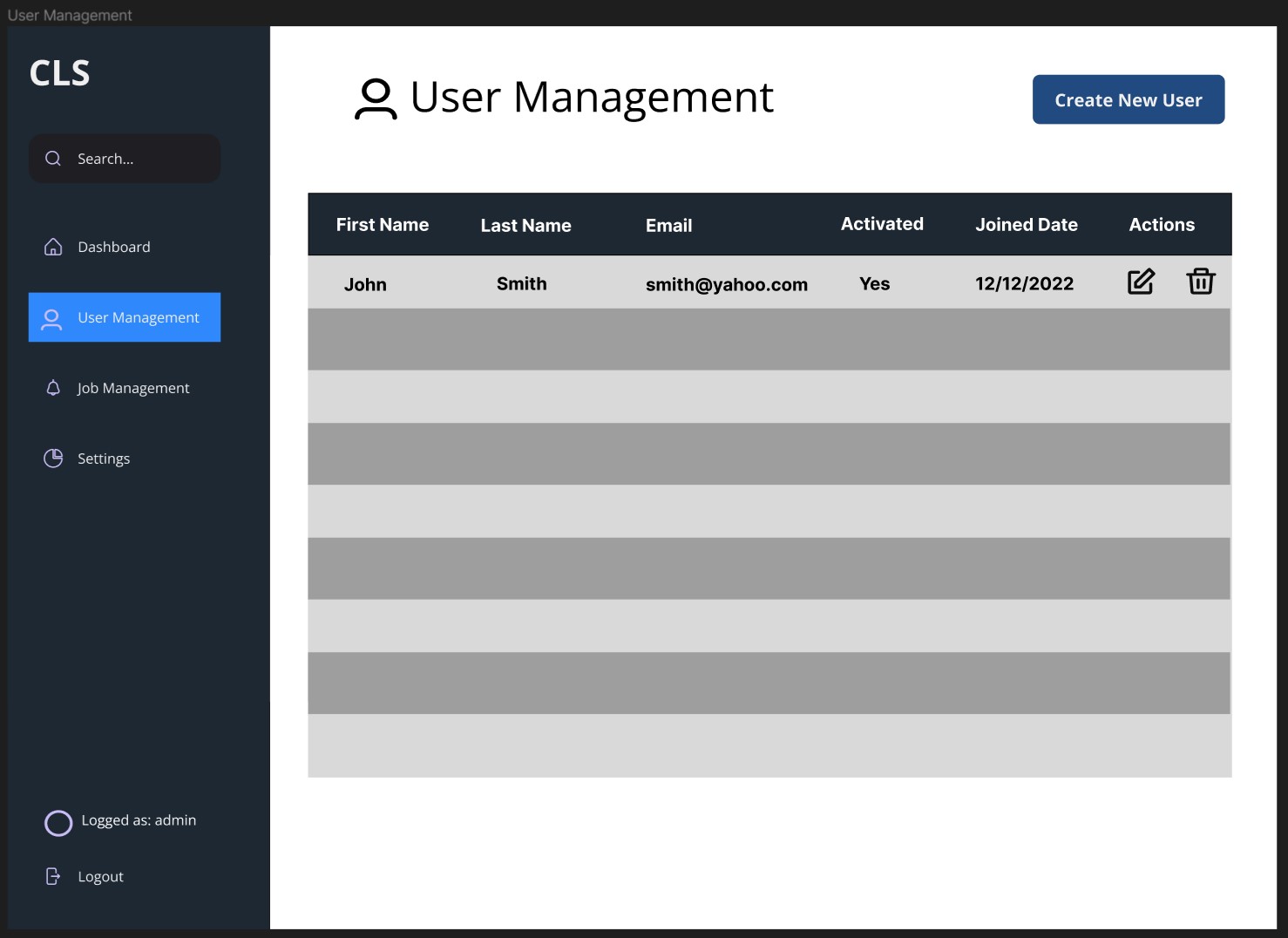
This page has two fields where user can type email and password. If all data are valid the page is redirected to the view of logged user. Dashboards are different depended on who is logged in crew member or admin.

Admin Account

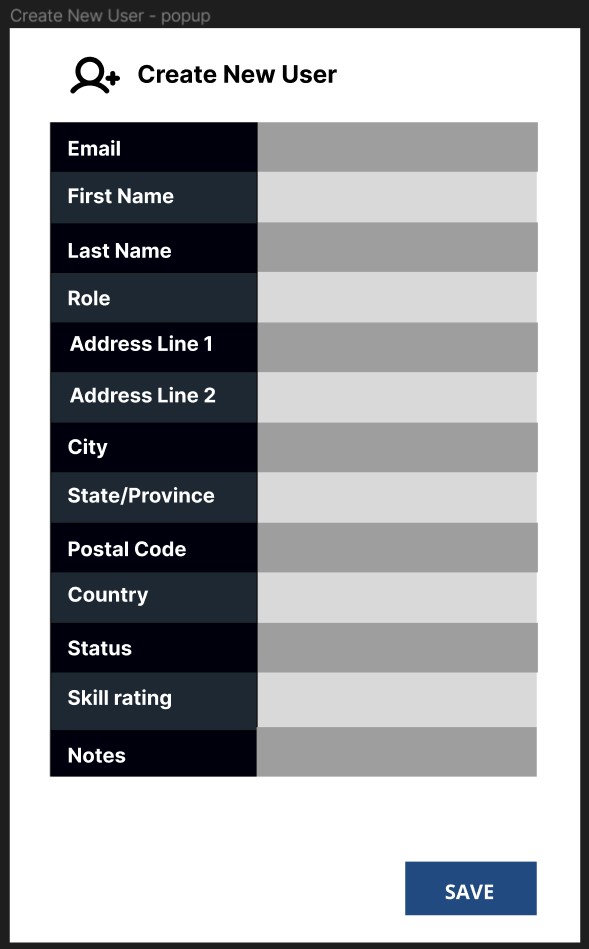
The design was created in Figma. Figma is a simple drag and drop software for developing user interfaces.

This dashboard will be appeared ones the admin will be logged in to his/her account. In future development will be decided what the notifications will to be.



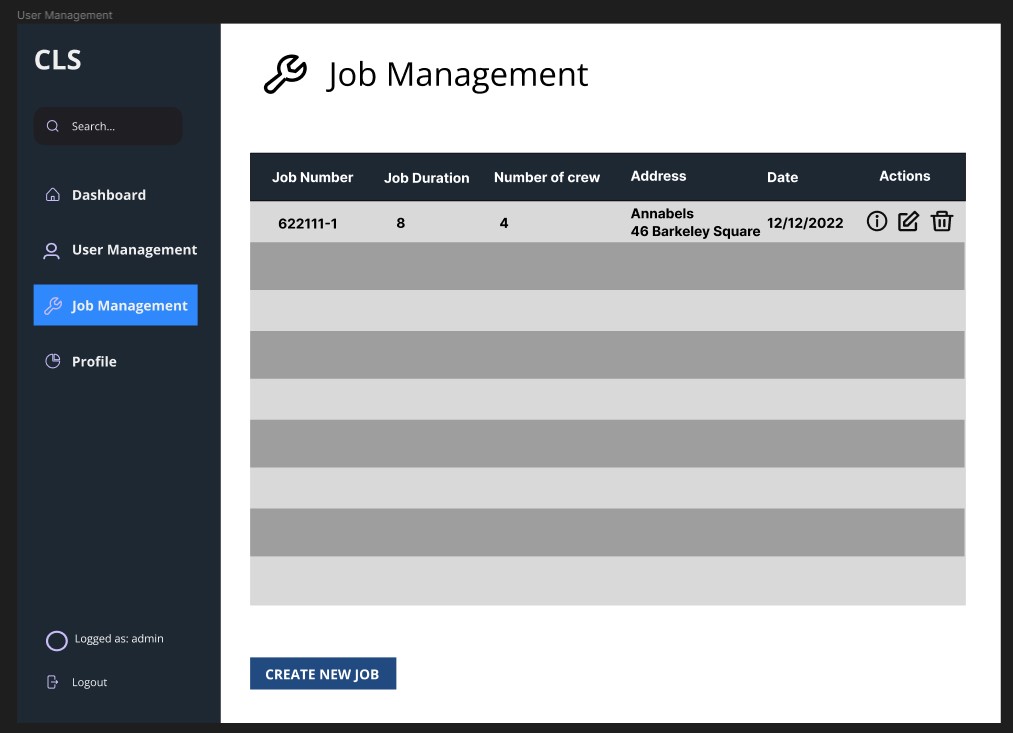
*Figure 11 Admin's dashboard*

*Figure 13 User management view*

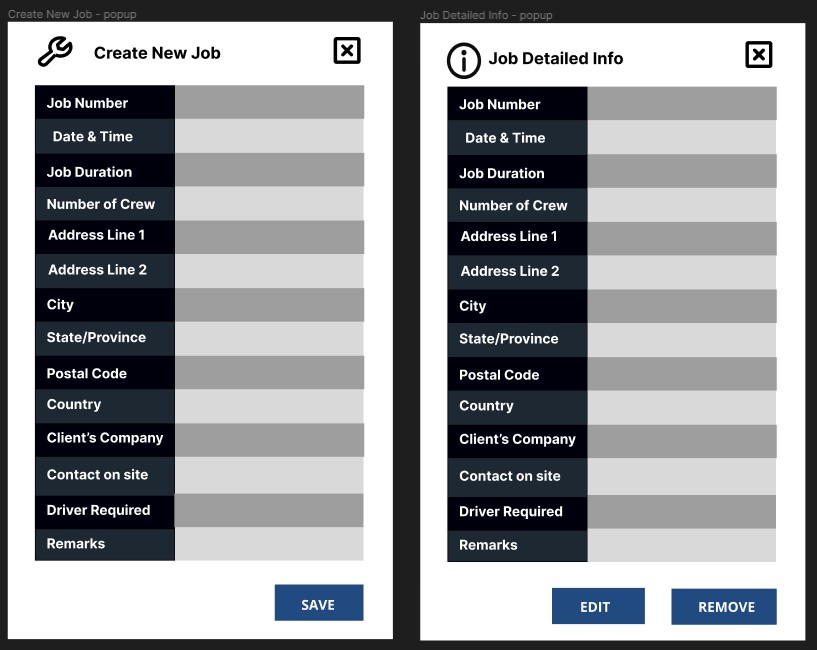


The view on the right-hand site is to create a user. The admin must click on the button “create new user” after that the window will pop up with a form to fill up. The edit view will be similar to it. The edit view will be popping up ones the button is clicked. The button is positioned in the list of users. The edit view will come up with the details from the database.

*Figure 12 Create New User view >*



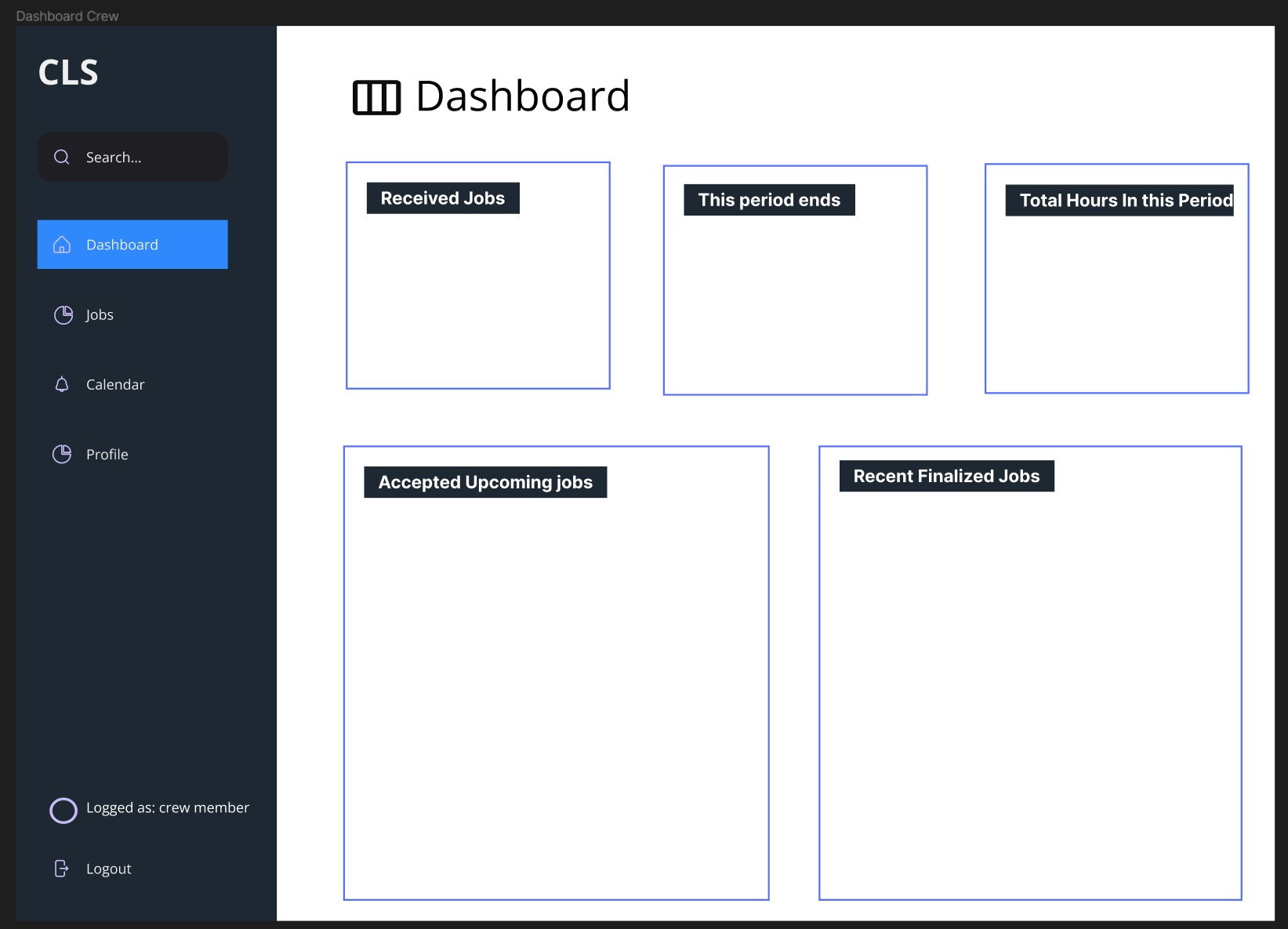
*Figure 15 Job Management View*



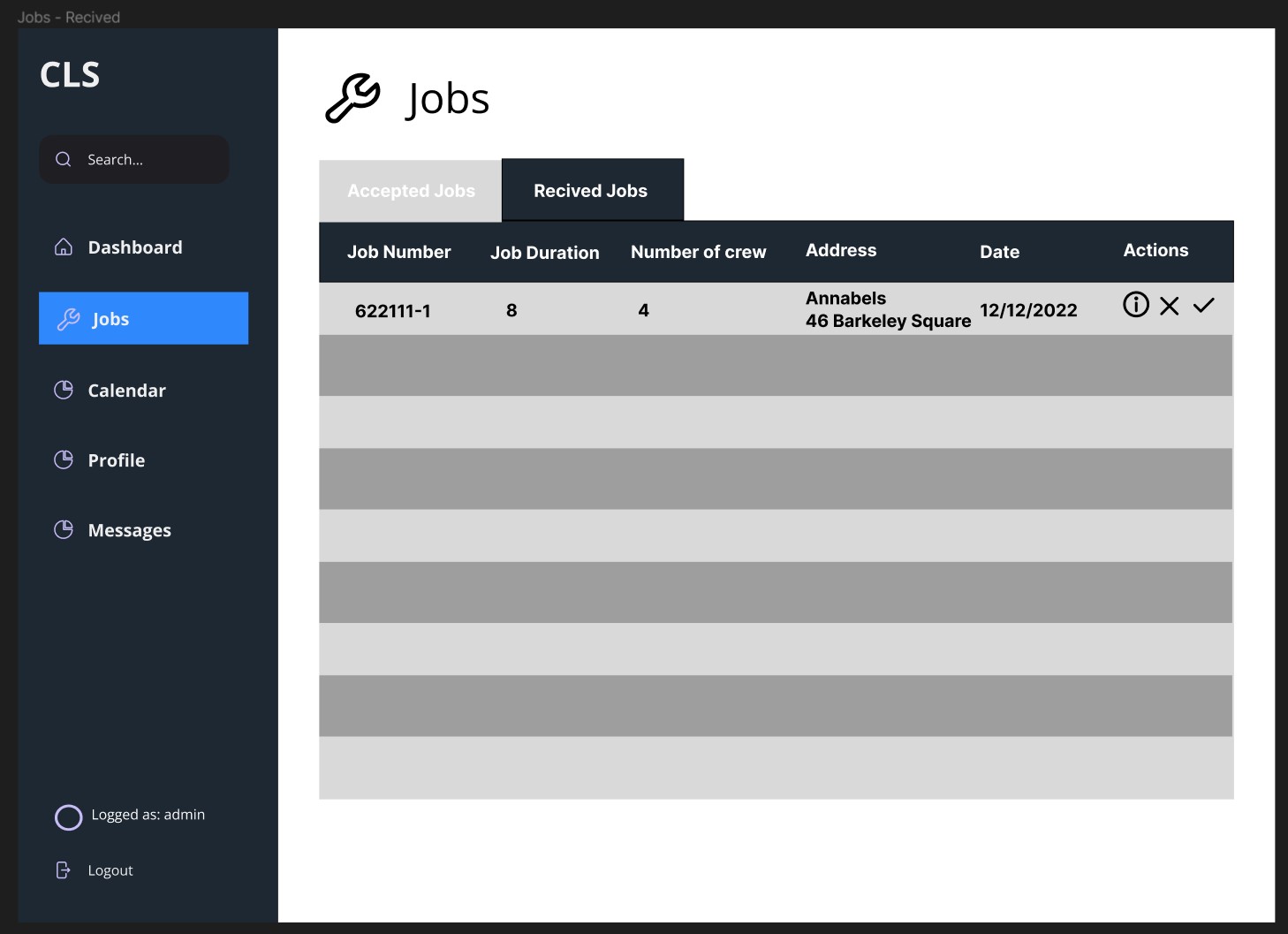
*Figure 14 New Job Form and Job Detailed Info popup windows*

Crew Account

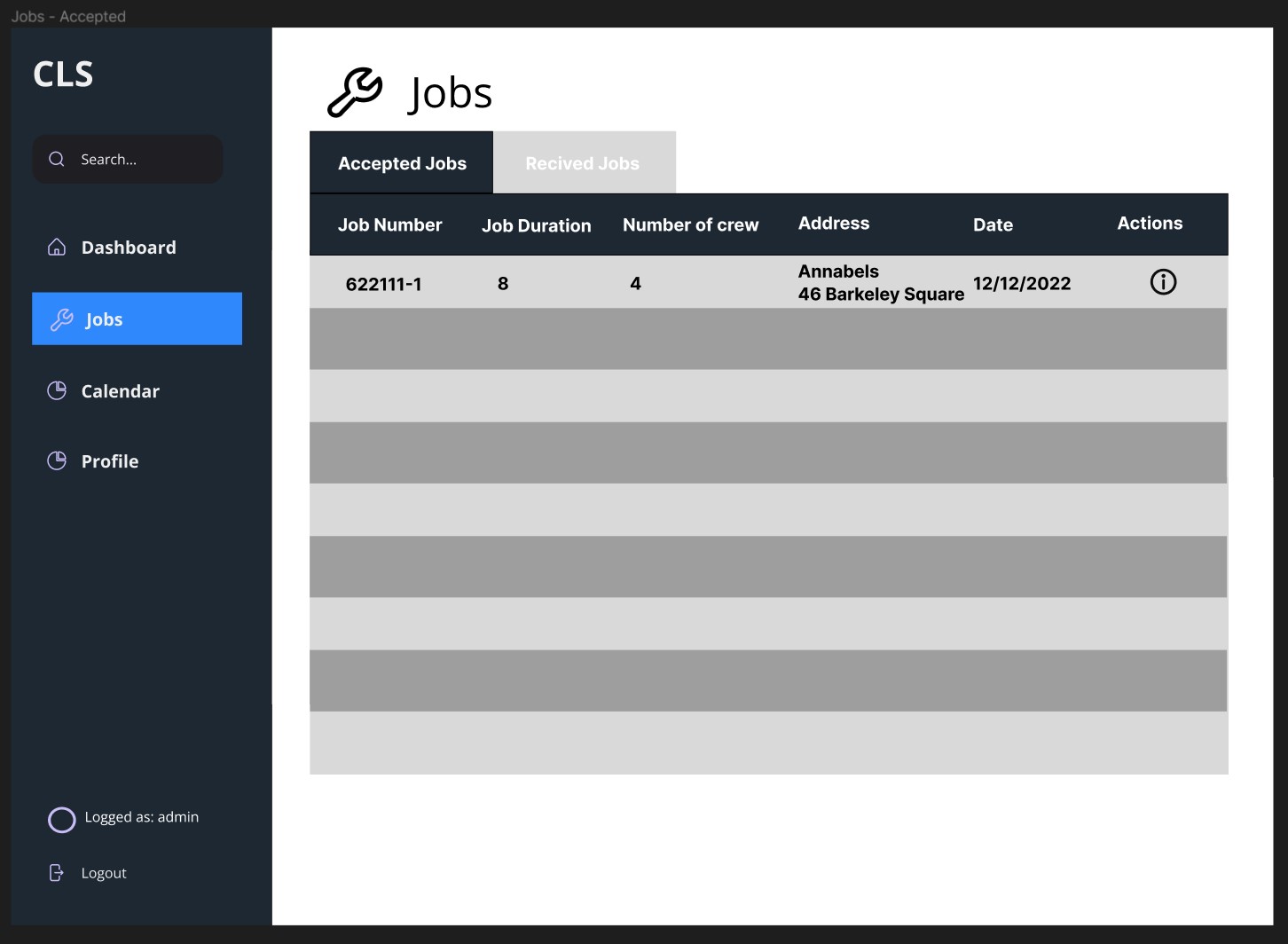
This dashboard shows the overall about the completed and upcoming jobs, also about what crew member gained. Hours and pay.



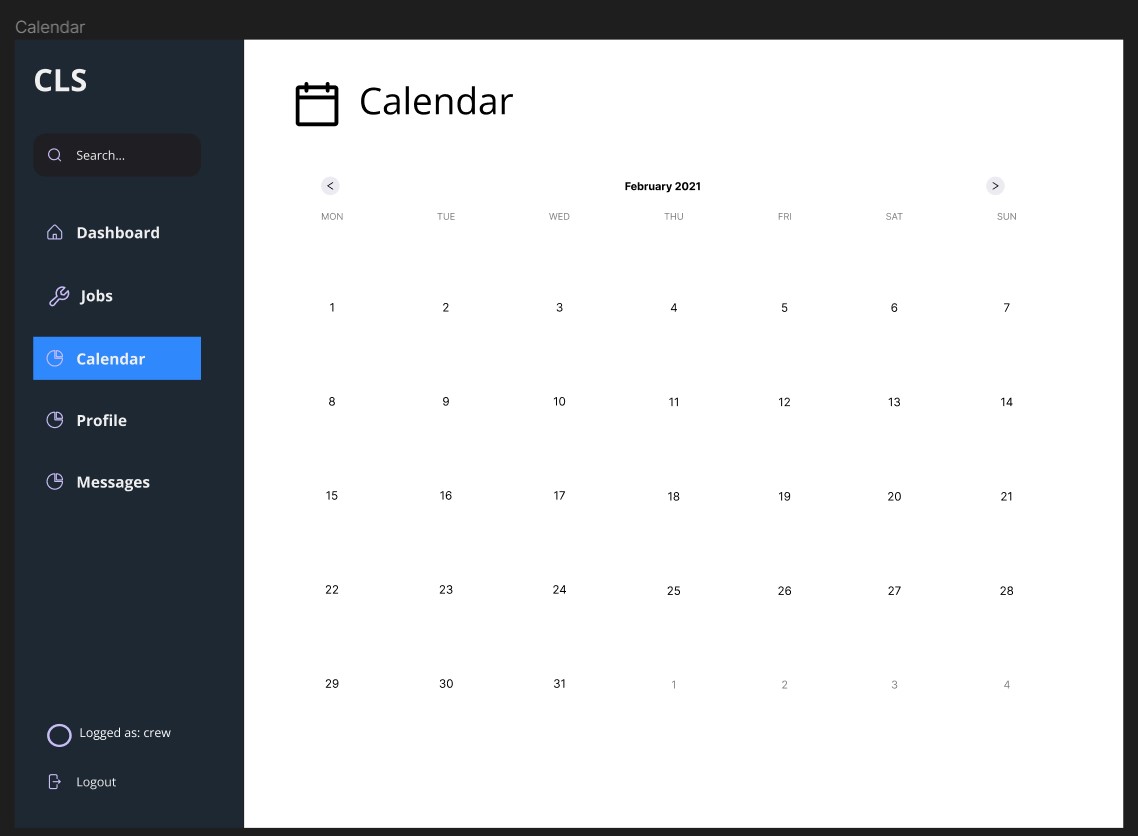
*Figure 16 Dashboard of crew member*



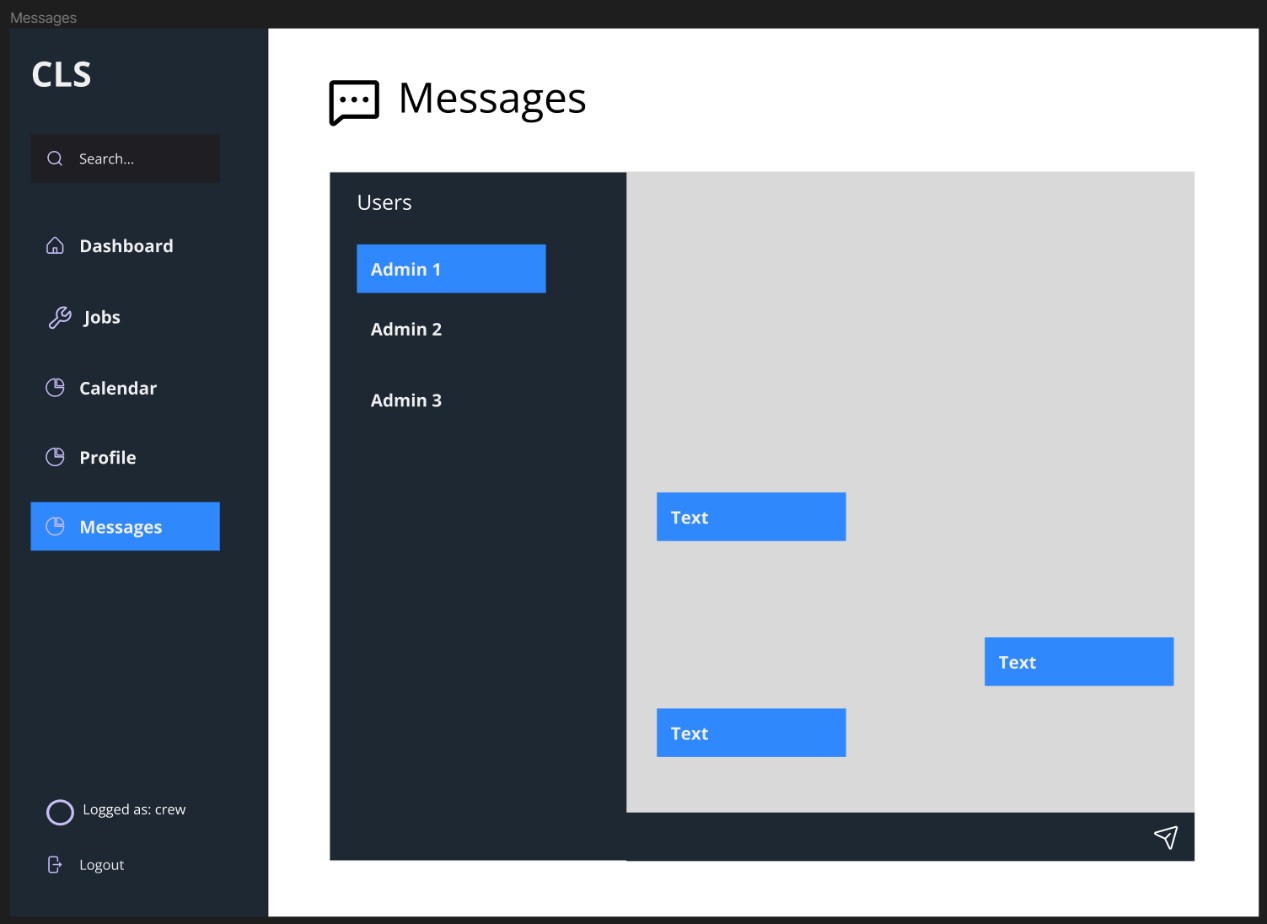
*Figure 18 Received Jobs from Admin to accept or decline*



*Figure 17 Accepted Jobs / Upcoming jobs for logged crew member*



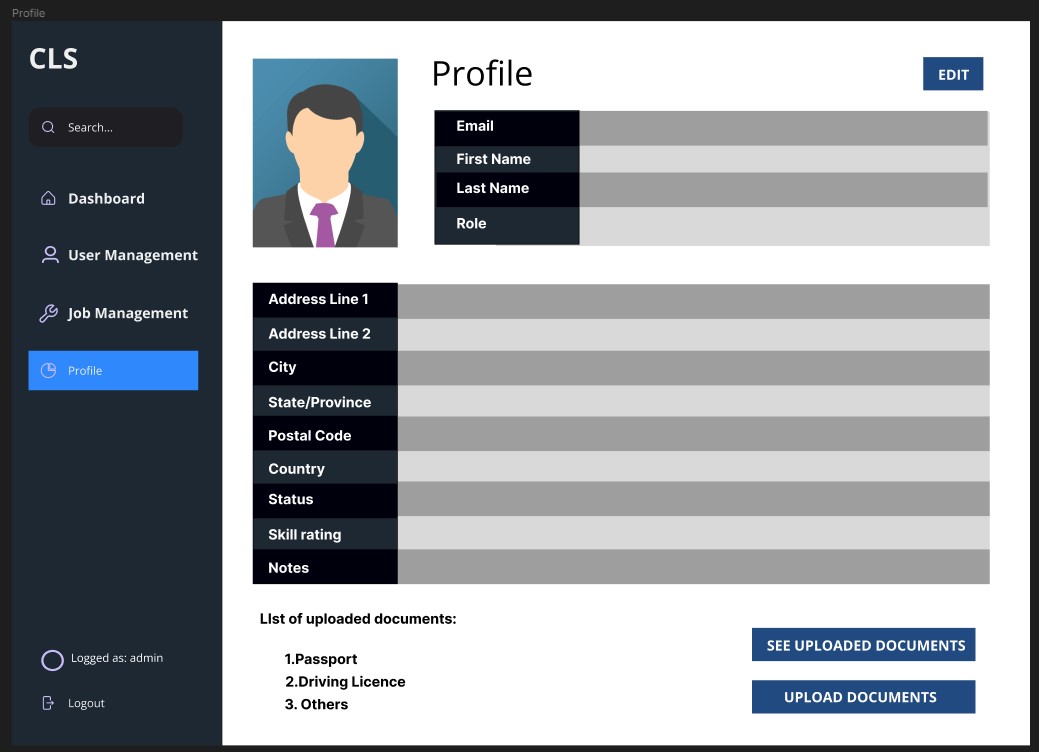
*Figure 19 Planning calendar for each crew member*



*Figure 20 Conversations between admins and crew*

User’s profile

The navbar of this view will be different for crew or admin but the profile section will be the same.

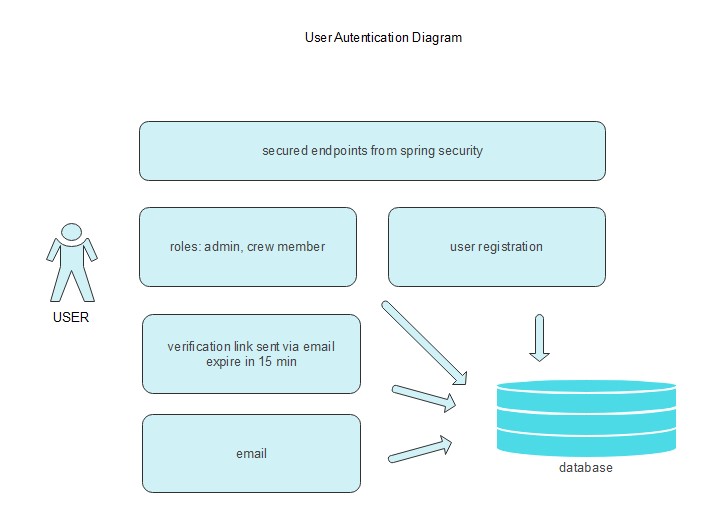


*Figure 21 Profile View*

# Chapter 5. Summary and Future Work

The code has been already written with following functionalities:

Complete Login – the system recognizes who is logging in crew member nor admin. The system using JWT which is Json Web Token. JWT is a string which is saved on client’s computer to be able to create a session with all data for user. Its not necessary to load data from database to see if the credentials are valid.



*Figure 22 User Authentication diagram*

Create a user – the user can be created by the admin. There is appropriate form to do that.

Confirmation Email – ones the user is created the system sends the confirmation letter for the provided email by admin. Ones the user clicks on the link provided in the email the account became activated. This all has been made for the security reasons.

What needs to still be done is to restructure the design which has been attached in this report. The design which is developed is not clear and there is a lot to make. The idea is to substitute what has been already done and apply new template in React. Is easer to find a template for the system than creating it from scratch.

# References

1. Diawati, Prety & Paramarta, Vip & Pitoyo, Djoko & Fitrio, Tomy & Mahrani, Wiyati. (2019). Challenges of Implementing an Employee Management System for Improving Workplace

Management Effectiveness. Available at [https://www.researchgate.net/profile/TomyFitrio/publication/338216184\_Challenges\_of\_Implementing\_an\_Employee\_Management\_S ystem\_for\_Improving\_Workplace\_Management\_Effectiveness/links/5e07f14a4585159aa4a](https://www.researchgate.net/profile/Tomy-Fitrio/publication/338216184_Challenges_of_Implementing_an_Employee_Management_System_for_Improving_Workplace_Management_Effectiveness/links/5e07f14a4585159aa4a25ba6/Challenges-of-Implementing-an-Employee-Management-System-for-Improving-Workplace-Management-Effectiveness.pdf)

[25ba6/Challenges-of-Implementing-an-Employee-Management-System-for-Improving-](https://www.researchgate.net/profile/Tomy-Fitrio/publication/338216184_Challenges_of_Implementing_an_Employee_Management_System_for_Improving_Workplace_Management_Effectiveness/links/5e07f14a4585159aa4a25ba6/Challenges-of-Implementing-an-Employee-Management-System-for-Improving-Workplace-Management-Effectiveness.pdf)

[Workplace-Management-Effectiveness.pdf](https://www.researchgate.net/profile/Tomy-Fitrio/publication/338216184_Challenges_of_Implementing_an_Employee_Management_System_for_Improving_Workplace_Management_Effectiveness/links/5e07f14a4585159aa4a25ba6/Challenges-of-Implementing-an-Employee-Management-System-for-Improving-Workplace-Management-Effectiveness.pdf) . Last accessed 13 December 2022

1. Julie Bulmash, “Human Resource Management and Technology”, Chapter 3

# Bibliography

Ian Somerville 2014 Software Engineering 10th ed. Essex: Pearson.

Artemij Fedosejev. (2015). *React.js Essentials*. Packt Publishing.

Somnath Musib. (2022). *Spring Boot in Practice*. Manning Publications.

Pinyodulyajet, S., 2002. E-Logistic application.