**Table of Contents**

[**[YOURPROJECT] REQUIREMENTS SPECIFICATION** **1**](#_heading=h.19c6y18)

[**VERSION 1.0** **1**](#_heading=h.3tbugp1)

[**APRIL 19, 2021** **1**](#_heading=h.28h4qwu)

[**1.**](#_heading=h.nmf14n"﷟HYPERLINK "bookmark://_heading=h.nmf14n) ﷟HYPERLINK "bookmark://\_heading=h.nmf14n"**EXECUTIVE SUMMARY** **3**

[1.1](#_heading=h.37m2jsg"﷟HYPERLINK "bookmark://_heading=h.37m2jsg) ﷟HYPERLINK "bookmark://\_heading=h.37m2jsg"Project Overview 3

[1.2](#_heading=h.1mrcu09"﷟HYPERLINK "bookmark://_heading=h.1mrcu09) ﷟HYPERLINK "bookmark://\_heading=h.1mrcu09"Purpose and Scope of this Specification 3

[**2.**](#_heading=h.46r0co2"﷟HYPERLINK "bookmark://_heading=h.46r0co2) ﷟HYPERLINK "bookmark://\_heading=h.46r0co2"**PRODUCT/SERVICE DESCRIPTION** **3**

[2.1](#_heading=h.2lwamvv"﷟HYPERLINK "bookmark://_heading=h.2lwamvv) ﷟HYPERLINK "bookmark://\_heading=h.2lwamvv"Product Context 3

[2.2](#_heading=h.111kx3o"﷟HYPERLINK "bookmark://_heading=h.111kx3o) ﷟HYPERLINK "bookmark://\_heading=h.111kx3o"User Characteristics 3

[2.3](#_heading=h.3l18frh"﷟HYPERLINK "bookmark://_heading=h.3l18frh) ﷟HYPERLINK "bookmark://\_heading=h.3l18frh"Assumptions 3

[2.4](#_heading=h.206ipza"﷟HYPERLINK "bookmark://_heading=h.206ipza) ﷟HYPERLINK "bookmark://\_heading=h.206ipza"Constraints 3

[2.5](#_heading=h.4k668n3"﷟HYPERLINK "bookmark://_heading=h.4k668n3) ﷟HYPERLINK "bookmark://\_heading=h.4k668n3"Dependencies 4

[**3.**](#_heading=h.2zbgiuw"﷟HYPERLINK "bookmark://_heading=h.2zbgiuw) ﷟HYPERLINK "bookmark://\_heading=h.2zbgiuw"**REQUIREMENTS** **4**

[3.1](#_heading=h.1egqt2p"﷟HYPERLINK "bookmark://_heading=h.1egqt2p) ﷟HYPERLINK "bookmark://\_heading=h.1egqt2p"Functional Requirements 5

[3.2](#_heading=h.3ygebqi"﷟HYPERLINK "bookmark://_heading=h.3ygebqi) ﷟HYPERLINK "bookmark://\_heading=h.3ygebqi"Non-Functional Requirements 5

[*3.2.1*](#_heading=h.2dlolyb"﷟HYPERLINK "bookmark://_heading=h.2dlolyb) ﷟HYPERLINK "bookmark://\_heading=h.2dlolyb"*Product Requirements* *5*

[**3.2.1.1**](#_heading=h.44sinio"﷟HYPERLINK "bookmark://_heading=h.44sinio) ﷟HYPERLINK "bookmark://\_heading=h.44sinio"**User Interface Requirements** 6

[**3.2.1.2**](#_heading=h.2jxsxqh"﷟HYPERLINK "bookmark://_heading=h.2jxsxqh) ﷟HYPERLINK "bookmark://\_heading=h.2jxsxqh"**Usability** 6

[**3.2.1.3**](#_heading=h.z337ya"﷟HYPERLINK "bookmark://_heading=h.z337ya) ﷟HYPERLINK "bookmark://\_heading=h.z337ya"**Efficiency** 6

[3.2.1.3.1](#_heading=h.3j2qqm3"﷟HYPERLINK "bookmark://_heading=h.3j2qqm3) ﷟HYPERLINK "bookmark://\_heading=h.3j2qqm3"Performance Requirements 6

[3.2.1.3.2](#_heading=h.1y810tw"﷟HYPERLINK "bookmark://_heading=h.1y810tw) ﷟HYPERLINK "bookmark://\_heading=h.1y810tw"Space Requirements 6

[**3.2.1.4**](#_heading=h.4i7ojhp"﷟HYPERLINK "bookmark://_heading=h.4i7ojhp) ﷟HYPERLINK "bookmark://\_heading=h.4i7ojhp"**Dependability** 6

[**3.2.1.5**](#_heading=h.2xcytpi"﷟HYPERLINK "bookmark://_heading=h.2xcytpi) ﷟HYPERLINK "bookmark://\_heading=h.2xcytpi"**Security** 7

[*3.2.2*](#_heading=h.sqyw64"﷟HYPERLINK "bookmark://_heading=h.sqyw64) ﷟HYPERLINK "bookmark://\_heading=h.sqyw64"*Organizational Requirements* *7*

[**3.2.2.1**](#_heading=h.3whwml4"﷟HYPERLINK "bookmark://_heading=h.3whwml4) ﷟HYPERLINK "bookmark://\_heading=h.3whwml4"**Environmental Requirements** 7

[**3.2.2.2**](#_heading=h.2bn6wsx"﷟HYPERLINK "bookmark://_heading=h.2bn6wsx) ﷟HYPERLINK "bookmark://\_heading=h.2bn6wsx"**Operational Requirements** 7

[**3.2.2.3**](#_heading=h.qsh70q"﷟HYPERLINK "bookmark://_heading=h.qsh70q) ﷟HYPERLINK "bookmark://\_heading=h.qsh70q"**Development Requirements** 7

[*3.2.3*](#_heading=h.3cqmetx"﷟HYPERLINK "bookmark://_heading=h.3cqmetx) ﷟HYPERLINK "bookmark://\_heading=h.3cqmetx"*External Requirements* *7*

[**3.2.3.1**](#_heading=h.1pxezwc"﷟HYPERLINK "bookmark://_heading=h.1pxezwc) ﷟HYPERLINK "bookmark://\_heading=h.1pxezwc"**Regulatory Requirements** 7

[**3.2.3.2**](#_heading=h.49x2ik5"﷟HYPERLINK "bookmark://_heading=h.49x2ik5) ﷟HYPERLINK "bookmark://\_heading=h.49x2ik5"**Ethical Requirements** 7

[**3.2.3.3**](#_heading=h.2p2csry"﷟HYPERLINK "bookmark://_heading=h.2p2csry) ﷟HYPERLINK "bookmark://\_heading=h.2p2csry"**Legislative Requirements** 7

[3.2.3.3.1](#_heading=h.147n2zr"﷟HYPERLINK "bookmark://_heading=h.147n2zr) ﷟HYPERLINK "bookmark://\_heading=h.147n2zr"Accounting Requirements 7

[3.2.3.3.2](#_heading=h.3o7alnk"﷟HYPERLINK "bookmark://_heading=h.3o7alnk) ﷟HYPERLINK "bookmark://\_heading=h.3o7alnk"Security Requirements 7

[3.3](#_heading=h.1rvwp1q"﷟HYPERLINK "bookmark://_heading=h.1rvwp1q) ﷟HYPERLINK "bookmark://\_heading=h.1rvwp1q"Domain Requirements 7

[**4.**](#_heading=h.4bvk7pj"﷟HYPERLINK "bookmark://_heading=h.4bvk7pj) ﷟HYPERLINK "bookmark://\_heading=h.4bvk7pj"**USER SCENARIOS/USE CASES** **7**

# Executive Summary

## Project Overview

Technology has evolved and has make people life easier in many different fields. Using technology in daily tasks of humans life is a convenient and efficient way. One field in which technology has begin to be used a lot is transportation project focuses on developing an online booking platform for booking train tickets. In this project we aim to make passengers experience better by using technology.

Passengers in many places are used with buying tickets for train or bus physically. And most of them are not happy with this process. As they do no not like waiting in queues or wasting time to go agencies to buy them. With our online booking platform, passengers will have the convenience of booking tickets anytime and from anywhere, eliminating the need to physically visit ticket counters or agencies. This level of accessibility not only saves valuable time but also offers the flexibility to plan journeys efficiently.

By streamlining the ticketing process, our platform minimizes waiting times and eliminates queues, ensuring a better booking experience.

Moreover, our platform prioritizes customer service by offering a range of amenities, including ticket modifications, and timely communication regarding updates or cancellations,checkout. This personalized approach aims to elevate the overall customer experience, satisfaction and loyalty.

In summary, our aim is to create an online booking platform where passengers can book their tickets whenever they want and by benefiting from facilities that this program offers. This represents a significant step forward in modernizing train ticket reservations. By using the power of technology, we aim to make booking tickets easier, more efficient, and ultimately, more enjoyable for passengers.

***Purpose and Scope of this Specification***

The main purpose of the project is to make passengers experience better. By using the opportunities that technology offers we will create a program that will help passengers to book their tickets online and this will be a better, more efficient and convenient way compared with the other booking methods.

Intended Audience: The intended audience for this specification includes:

* Passengers
* Employee
* Admin

The web application will offer passengers possibility to book at anytime his train ticket ,for any destination available etc.Employee can add passangers,update passengers ,see history .And admin which is the head and it’s duty is to add employee ,modify them.Admin will also have all the credentials that employee has.

# Product/Service Description

Our Online train ticket booking is a user-friendly platform designed to streamline the process of booking train tickets, making the process faster and easier.

It provides convenience and efficiency to passengers, while also facilitating the management and surveillance of the booking procedure.

The streamlined booking process and real-time updates enhance effectiveness and minimize nuisance. Any changes made by the staff/managers will be instantly accessible by all users.

With personalized user accounts, travelers can easily manage their bookings, view train schedules, and access relevant travel information. Users can conveniently reserve train tickets anytime, anywhere, with just a few clicks.

Employees can easily access passenger and ticket information, while also being able to make necessary changes.

Managers have a broader scope of control over the system, being allowed to access more classified information, like finance and train registering.

## User Characteristics

There are 3 types of users in this system.

**Passenger/Client**

* Visits the website to book train tickets
* Creates an account to be able to access bookings
* Views available trains and their respective information, including departure time, routes, fare
* Cancels bookings
* Makes ticket payment
* Changes personal information
* Has an account history that provides a comprehensive record of past and upcoming bookings for easy access
* May also include seat selection / number of available seats

**Employee**

* Cannot register on their own
* Can login with credentials provided by the employer
* Has access to general information like train and passenger information, booking history
* Can register new passengers
* Can view/update/delete passengers
* Can approve and manage tickets
* May have the access to change train info (route/departure)

**Manager/Administrator**

* Can login with their own credentials
* Has all the accessibility and functionality of the employee
* Has access to employee information
* Can add/delete/update employees
* Can view financial information
* Can add/manage trains

## Assumptions

* It is assumed that the payment is made with the necessary security and procedures. The bank account info is not included in the payment procedure, nor the credit card transactions.
* It is assumed that before registering a new employee, the necessary documents have been reviewed by the corresponding experts. The admin oversees registering the employees, with their basic account information.
* It is assumed that there are clients who cannot register by themselves, or who have bought physical tickets, so their registration can be done by employees/manager.
* It is assumed that, despite having their different roles, the manager can access all functions that are available to the employees, so that they have broader control over the system. There may be cases that the manager intervention could be needed.
* It is assumed that the manager has access to train information and previous decisions made by the train station board and is given the role of adding and registering new trains to the system.
* It is assumed that the employee is provided with the necessary precise and pre-determined information to modify train routes/departures

## Constraints

* The program may need to run on specific hardware platforms such as desktop computers, which could limit its accessibility.
* The users may have different levels of technical knowledge and experience, which could affect the usability of the program, even though it will be developed in a way that is as easily understandable as possible.
* Since the program is a web application, a stable internet connection is needed to access it.
* The system may be subject to regulations or guidelines governing the use of personal data, which could affect the security and privacy of the program.
* The program may need to integrate with third-party software or services to enable payment processing / online transactions, which could affect the compatibility of the program.

## Dependencies

The roles in this program are linked to each other.

* The client can make the payment for a reservation, but the employee or manager must accept it.
* The employee cannot login if they do not have the credentials provided by the employer.

# Requirements

* Describe all system requirements in enough detail for designers to design a system satisfying the requirements and testers to verify that the system satisfies requirements.
* Organize these requirements in a way that works best for your project. See Appendix DAppendix D, Organizing the Requirements for different ways to organize these requirements.
* Describe every input into the system, every output from the system, and every function performed by the system in response to an input or in support of an output. (Specify what functions are to be performed on what data to produce what results at what location for whom.)
* Each requirement should be numbered (or uniquely identifiable) and prioritized.

See the sample requirements in Functional Requirements, and System Interface/Integration, as well as these example priority definitions:

**Priority Definitions**

The following definitions are intended as a guideline to prioritize requirements.

* Priority 1 – The requirement is a “must have” as outlined by policy/law
* Priority 2 – The requirement is needed for improved processing, and the fulfillment of the requirement will create immediate benefits
* Priority 3 – The requirement is a “nice to have” which may include new functionality

It may be helpful to phrase the requirement in terms of its priority, e.g., "The value of the employee status sent to DIS **must be** either A or I" or "It **would be nice** if the application warned the user that the expiration date was 3 business days away". Another approach would be to group requirements by priority category.

* A good requirement is:
* Correct
* Unambiguous (all statements have exactly one interpretation)
* Complete (where TBDs are absolutely necessary, document why the information is unknown, who is responsible for resolution, and the deadline)
* Consistent
* Ranked for importance and/or stability
* Verifiable (avoid soft descriptions like “works well”, “is user friendly”; use concrete terms and specify measurable quantities)
* Modifiable (evolve the Requirements Specification only via a formal change process, preserving a complete audit trail of changes)
* Does not specify any particular design
* Traceable (cross-reference with source documents and spawned documents).

## Functional Requirements

| **Req#** | **Requirement** | **Comments** | **Priority** | **Date Rvwd** | **SME Reviewed / Approved** |
| --- | --- | --- | --- | --- | --- |
| FR\_01 | The software should have different views for different user levels | Different view for passenger,admin and employee | 2 |  |  |
| FR\_02 | The user,admin,employee account have to be secured by password | We will ensure ethical practices by employing password hashing before storing them in the database. | 1 |  |  |
| FR\_03 | After user/passenger have created it’s profile .He can modify it’s personal details and change password. | User can edit it’s username, birthday date,surname,email,  password ,phone no ,put a photo etc. | 1 | 7/13/04 | Bob Dylan, Mick Jagger |
| FR\_04 | The passenger see all available trains and also search train . | Passenger can check all available trains in platform and details for each and also search by name ,time, price train no,route,depart,arrival,dep time, total passengers of train. |  |  |  |
| FR\_05 | The system should provide the passenger to book the train. And after booking ,he can cancel it. | Passenger can cancel the train after booking it. |  |  |  |
| FR\_06 | The passenger can search for it’s ticket after booking it. | If passenger have booked two or more tickets. It can search about a specific ticket he want at search bar. |  |  |  |
| FR\_07 | Passenger after booking ticket, will check out and after it need to confirm the payment. | At check out passenger needs to put the code. After it it needs to confirm the payment. |  |  |  |
| FR\_08 | Passenger have the option to print also the ticket. | Passenger can print/download a ticket as a pdf file. |  |  |  |
| FR\_09 | Passenger can log out after he have completed the tasks he want. |  |  |  |  |
| FR\_10 | The employee can see number of passengers ,trains, booked tickets and pending tickets. | Employee can see after log in in dashboard total number of passengers and total number of available trains. Also number of booked tickets and pending one in order to approve later. |  |  |  |
| FR\_11 | Employee can view, update it’s profile, change password and put a profile avatar/photo. | Employee has the ability to change personal details like name, phone number,address,username,  Last name, password .And put a profile picture. |  |  |  |
| FR\_12 | Employee can update and add a new train. | Employee has the credits to add a new train by putting his a name,number,route,departure destination,arrival,departure time, number of passengers and train fare. Moreover he can update by deleting or modifying train details. He can also view it. |  |  |  |
| FR\_13 | Employee have credits to add a new passenger and to manage passengers. | Employee can add a new passenger by putting it’s personal details. And can manage them by updating personal details, deleting them. |  |  |  |
| FR\_14 | Employee has credits also to approve tickets,modifythem,delete them. | Employee can see from dashboard approved tickets and pending tickets it can approve the pending ones. Also can delete,modify,view tall tickets. |  |  |  |
| FR\_15 | Employee can log out. |  |  |  |  |
| FR\_16 | Admin can view from dashboard number of passangers,trains,reservation,employees,adminstators and number of approved tickets and pending tickets. | In dashboard admin can have access to all of these. |  |  |  |
| FR\_17 | Admin can view it’s profile and can update it. | Admin has access to change all his personal details ,put a profile avatar and change password |  |  |  |
| FR\_18 | Admin has credits to add a new train and manage them. | As employee also admin has ability to add a new train with all it’s specifics and manage them by editing ,deleting them. |  |  |  |
| FR\_19 | Admin has credits to add a new passenger and manage them. | As employee also admin has ability to add a new passenger with all it’s specifics and manage them by editing ,deleting them. |  |  |  |
| FR\_20 | Admin has access to a add a new employee and manage them. | Admin can put a new employee in system and also filling all it’s personal details +password. Also Admin can update employees information and delete anyone of them |  |  |  |
| FR\_21 | Admin has access also at payments. | Adim can check all pending payments and all paid reservations. |  |  |  |
| FR\_22 | Admin can log out |  |  |  |  |

## Non-Functional Requirements

### Product Requirements

* + Requirements which specify that the delivered product must behave in a particular way e.g. execution speed, reliability, etc.

#### **User Interface Requirements**

**Home interface:**

Menu Bar where you can select to :

* Admin Login
* Employee Login
* Make registration
* Home

**Registration Interface:**

* A form is open and at top of form is the logo of application
* User need to fill the form with *<Name><last name><phone number><address><email><password>*
* After it you can click register and your dates are validate and account is created.

**Log In for Passenger:**

* When this page is open a form is open
* At top of form there is logo of application
* If you already have an account you need to put *<email>* and *<password>.*You can choose checkbox Rember me to remember you password next time .When the button is pressed the information is validated and the respective interface is shown to user. Or an error notification indicating a wrong username or password will be shown.
* If you have forgotten you password you can click *on <forget password>,*then another interface for reset your password will be open
* If you do not have an account you will click on register button which will lead you to register interface

**Forget Password Interface:**

* A form is open and at top of form is the logo of application
* User need to fill the form *with <email>* and after you can reset you password.

**Employee Log in interface:**

* A form is open and at top of form is the logo of application
* User need to fill the form with <email><password>.You can choose checkbox Rember me to remember you password next time .When the button is pressed the information is validated and the respective interface is shown to employee. Or an error notification indicating a wrong username or password will be shown.
* If employee do not remember password can click on right to *<forgot password?>* and Forget Password Interface will be open.

**Admin Login interface:**

* A form is open and at top of form is the logo of application
* User need to fill the form with *<email><password>.*You can choose checkbox Rember me to remember you password next time .When the button is pressed the information is validated and the respective interface is shown to admin.Or an error notification indicating a wrong username or password will be shown.
* If employee do not remember password can click on right to *<forgot password?>* and Forget Password Interface will be open.

**Costumer Interface:**

Which will contain:g

* The header bar that will contain the name of customer and photo
* By clicking on profile picture section in header the passenger can go to it’s account where or personal inrfomation appear there and can Logout.
* Then at dashboard page it will be shown **Booked Train,Cancel booking ,Tickets** and a list with all ***avaible trains***.
* In Dashboard there is also customer profile ,which can **View** it, **can Update,Put Profile avatar** it and can also **change password.**In view menu it can see it’s profile.In update a form will be open and customer can change all personal details by updating the part of form we want to change *< name>,<last name>,<email>,<address>,<phone number>.*In Profile Avatar a form where you can upload a file will be open.In change password a from will be open where user need to put *<old password> and <new password> ,<confirm new password>*
* In Dashboard the user can have access to All available trains where it can se all available trains and all information for each train and .Moreover it has Search in which the customer can search about a specific train. The customer can search for that train by using as keyword any information related to the train(such as name,number,departure etc).Also in this page there is an option where passenger can chose how many entries he want 0,25,50 or 100.There is an icon to refer this part.And at the end of page there is also page number,previous and next option for pages of availability trains.
* Customer also in dashboard have Reserve train where it see all available trains and can book any train he want and cancel train where it can cancel the train.Also here there is a search box and box with option to select the number of entries .And at the end there is the option of previous and next page.
* Tickets where customer after reserving the train will do checkout by putting the number of card and after it he will confirm payments. Moreover it has also the option to print in pdf file the ticket at Print in dashboard.
* At the end of dashboard is Logout option.

**Employee Interface:**

Which will contain:

* The header bar that will contain the name of employee and photo.
* By clicking on profile picture section in header the employee can go to it’s account where or personal information appear there or can click on Logout.
* Then at dashboard page is like home page it will be shown number of passengers with icon on the left ,number of trains with icon of train on left, number of booked tickets with icon of ticket on left and number of pending ticket with its icon on left. In addition it has a table with all information for trains and another one for customers.
* In Dashboard there is also employe profile ,where every employee can **View** its profile, **can Update, Put Profile avatar** it and can also **change password.** In view menu it can see it’s profile. In update a form will be open and customer can change all personal details by updating the part of form we want to change *< name>,<last name>,<email>,<address>,<phone number>*.In Profile Avatar a form where you can upload a file will be open. In change password a from will be open where user need to put *<old password> and <new password> ,<confirm new password>*
* There is also Train in dashboard where an employee can add a Train by filling the form with detailed information about the train added .The employee will put *<Train name>.<Train Number>,<Train Route>,<Departure>,<Arrival>,<Departure Time>,<Number of passengers>,<Train Fare>.*In addition employee can Manage trains at Manage Trains. There it will appear the table with all trains available and employee can update any of them by clicking on update where a form with completed data will appear ,employee can change any data he wants. Employee can delete any train and can view it.Here employee can also search the train at Search box. Also in this page there is an option where employee can chose how many entries he want 0,25,50 or 100.There is an icon to refer this part. And at the end of page there is also page number, previous and next option for pages of availability trains.
* Same idea stands also for passengers .In dashboard there is a option for passenger where the customer can add passenger .When employee click on it a form with personal details of passenger will be open which need to be filled by employee. Moreover in Manage Passenger section employee will see a table with all passengers and their information. There employee can modify,delete and view passengers informations.Here employee can also search for a passenger at Search box. Also in this page there is an option where employee can chose how many entries he want 0,25,50 or 100.There is an icon to refer this part. And at the end of page there is also page number, previous and next option for pages of passengers.
* At Tickets in dashboard there are 3 section.One is for viewing approved tickets ,one for viewing pending ticktes where employee can approve them.And last one is for managing tickets where a table with all passengers and their info apperar.Employee can search them at search tool bar .Employee can modify their information by recompleting the form for passenger.Also he can delete any passenger ticket.
* At the end of dashboard is Logout option.

**Admin Interface:**

Which will contain:

* The header bar that will contain the icon of admin.
* By clicking on icon in the header the administrator can go to it’s account where or personal information appear there or can click on Logout option.
* Then at dashboard page is like home page it will be shown number of passengers with icon on the left ,number of trains with icon of train on left, number of reservations with its icon on left ,number of employees and administrators with employee icon and administrator icon respectively on left, and number of pending ticket with its icon on left.In addition it has a table with all information for trains and another one for passengers reservations and employees list.
* In Dashboard there is also employe profile ,where every employee can **View** its profile, **can Update,Put Profile avatar** it and can also **change password.**In view menu it can see it’s profile.In update a form will be open and customer can change all personal details by updating the part of form we want to change *< name>,<last name>,<email>,<address>,<phone number>.*In Profile Avatar a form where you can upload a file will be open.In change password a from will be open where user need to put *<old password> and <new password> ,<confirm new password>*
* There is also Train in dashboard where an employee can add a Train by filling the form with detailed information about the train added .The employee will put *<Train name>.<Train Number>,<Train Route>,<Departure>,<Arrival>,<Departure Time>,<Number of passengers>,<Train Fare>.*In addition employee can Manage trains at Manage Trains. There it will appear the table with all trains available and employee can update any of them by clicking on update where a form with completed data will appear ,employee can change any data he wants. Employee can delete any train and can view it.Here admin can also search the train at Search box. Also in this page there is an option where admin can chose how many entries he want 0,25,50 or 100.There is an icon to refer this part.And at the end of page there is also page number, previous and next option for pages for table trains.
* Same idea stands also for passengers .In dashboard there is a option for passenger where the customer can add passenger .When employee click on it a form with personal details of passenger will be open which need to be filled by employee. Moreover in Manage Passenger section employee will see a table with all passengers and their information. There employee can modify,delete and view passengers informations.Here employee can also search for a passenger at Search box. Also in this page there is an option where admin can chose how many entries he want 0,25,50 or 100.There is an icon to refer this part.And at the end of page there is also page number,previous and next option for table for passengers.
* Admin can have access also at employees. There is a section in dashboard names Employees where admin can add an employee by filling the form that appears with information for employee, He also puts the employee password. In Manage Employee section there is a box where adim can search for a specific employee and can also update the information of employee ,can delete it and can view.There is also a search box for searching for a specific employee. Also in this page there is an option where admin can chose how many entries he want 0,25,50 or 100.There is an icon to refer this part.And at the end of page there is also page number,previous and next option for table of employees.
* At Tickets in dashboard there are 3 section.One is for viewing approved tickets ,one for viewing pending ticktes where employee can approve them.And last one is for managing tickets where a table with all passengers and their info apperar.Employee can search them at search tool bar .Employee can modify their information by recompleting the form for passenger.Also he can delete any passenger ticket.
* In password resets section ,admin can view a table with approved changed passwords and a also a table with pending can also approve these. At both pages there is a search box where admin can search by user email.
* In Payment Section admin can view paid tickets and amount of money from them. Number of pending tickets,number of approved tickets and number of reservations.Each of these have their respectively their icons on the left.In addition there is also a pie in Tickets Payment status which illustrates the number of paid reservation and pending payments .This can be printed, save as jpg or png .
* At the end of dashboard is Logout option.

#### **Usability**

**Learnability:**

* **Staff:** Train staff members, including ticketing agents and administrators, should be able to become proficient in using the system within a short timeframe, typically a few hours of training.
* **Passengers:** The system should be intuitive and user-friendly for passengers, requiring minimal training. Clear prompts and instructions should guide users through the booking process seamlessly.
* **User Guide:** This documentation will serve as a comprehensive user guide for both staff and passengers of the train system, offering step-by-step instructions and explanations of features.
* **Error Handling:** In case of errors during the booking process, the system will display specific and detailed error messages to users, helping them understand what went wrong and how to rectify it.
* **User Responsibility:** Users are responsible for their own actions when using the system, including booking tickets, making modifications, and understanding the terms and conditions.

**Accessibility:**

* **User Registration:** Each user will have a personal username and password upon registration in the train ticket booking system.
* **Staff Access:** Ticketing agents, administrators, and other relevant employees associated with the train system will have access to the booking platform, allowing them to manage bookings, check availability, and perform administrative tasks.
* **Passenger Access:** Passengers using the train services will have access to the booking platform, enabling them to search for train schedules, book tickets, select seats, and make reservations.
* **Availability:** Registered users, including passengers and employees, will have the flexibility to access the booking platform at any time and from any location with internet access, ensuring convenience and accessibility for all users.
* The registered users can access the system at any time and from any place.

#### **Efficiency**

* Each operation will be fast and in real time.
* Once the users have learned the system they will be able to perform each operation within minutes.

**3.2.1.5 Memorability**

* The user interfaces are designed to be easy understandable by the user (pictures, icons, buttons, descriptions etc.)

**3.2.1.6 Errors**

* Each time sensitive data is entered in the system double check procedure is applied where the user confirms the entered data.
* For most of the possible error scenarios, informative messages will be prepared to be shown to the user, so they know what the problem is.

### Organizational Requirements

**3.2.2.1 Availability**

* The application will be available to everyone who owns a PC connected to the Internet, therefore any user can access their data anywhere.
* The application will be developed such that it will not be forced to encounter downtime since the data used is sensitive and contains financial information.

**3.2.2.2 Latency**

The latency of the web application will depend on:

* The speed/efficiency of the operations used to handle data/requests
* The number of users simultaneously accessing the system.
* Internet connection
* The capacity of the database

**3.2.2.3 Monitoring**

To ensure the system is running smoothly and efficiently, the following monitoring requirements must be met:

* User actions/payments must be checked and verified in real-time by the administrator and employees.
* In any case, when changing personal data, validation of fields will be used (Entering the account password before changing personal information, changing passwords will have a security step).
* For any apparent problem, the user will be informed with pop-up messages.

**3.2.2.4 Maintenance**

* In case of system malfunction, a system restart should help. If that does not help, then maybe the server is down, so the maintenance department should be contacted.
* The software should be designed with modular architecture to facilitate easy maintenance and updates.
* The software will be designed to provide useful error messages to help administrators and developers quickly troubleshoot and fix issues.

**3.2.2.5 Operations**

Some of the operations that will be provided to the users are:

* The users can log in and access their personal information anytime.
* Create, Read, Update, Delete Employees (Done by Administrator).
* Create, Read, Update, Delete Passenger accounts (Done by Administrator/Employees).
* Create, Read, Update, Delete Trains (Done by Administrator/Employees).
* Passengers can view personal reservation history
* Administrators/Employees can view overall reservation history
* Administrators can view balance/finances
* Employees/Administrators can approve ticket payments
* Passengers can book/cancel tickets and make payments
* Users can change password/personal information
* Passengers can print tickets

**3.2.2.6 Portability**

* The system will be web-based; therefore, it will operate the same regardless of the operating system.
* All you need is a computer or mobile phone and an Internet connection.
* The software should be designed to be easily portable to different platforms or environments.

### External Requirements

**3.2.3.1 Security**

The system contains personal data and payment information, which requires a certain degree of privacy. The information of each user will be protected by encrypted passwords. Also, the system has role-based access control, allowing each user to only have a specific scope of accessible information accordingly.

**3.2.3.2 Protection**

To protect the system from malicious/accidental access, modification, disclosure, destruction, or misuse we will take the following precautions:

* Encrypt the most sensitive information (passwords) using the hashing method to protect privacy.
* Each client will only see their personal data
* Managers/employees will only see the necessary client information
* Manager is responsible for checking the authenticity of the information that they upload on the system (employee/client/train information), hence the system is not responsible
* We will keep track of the activity of each user, such that in case of a problem the user will be held responsible.
* To make sure that some action is intentional and not accidental, most of the actions of the users are protected by a pop-up window to confirm that action.
* The system will validate each data for special characters and other specific conditions before inserting or updating in the database.

**3.2.3.3 Authorization and Authentication**

The Authorization and Authentication factors:

* The user authentication will be using username and password
* Authorization will be based on the user type. Each user will access only the respective information.
* If the user tries to log in with the wrong credentials a message will be shown to them.

## Domain Requirements

This system operates in the field of Transport. The goal of this project is to digitalize and make the way people can book their train tickets easier and faster. Another key point of the project is Data Security which is very important. Unauthorized users will not have access to passengers information .Only users who have appropriate permissions can have access. The ticketing system operates independently within the railway network and does not need to communicate with external systems.

**4. Software Design / Diagrams**

***4.1 Requirements Analysis***

**4.1.1 User Scenarios**

**4.1.1.1 User Scenarios List**

|  |  |  |
| --- | --- | --- |
| **Nr** | **Name** | **Description** |
| **US\_01** | User logs in | Users: administrator , employee and passengers log in using email and password |
| **US\_02** | Change password | Users: administrator , employee and passengers change their password |
| **US\_03** | Change the profile avatar | The user can change their avatar based on their preferences. |
| **US\_04** | Update their information | Users can update their information |
| **US\_05** | Passengers can view trains | The passengers can view the list of trains in system and search for a particular train. |
| **US\_06** | Passenger books a train | Passenger books a train from the list of trains |
| **US\_07** | Passenger cancel the bookings | Passenger can cancel their booked train. |
| **US\_08** | Checkout tickets | Passengers can checkout tickets |
| **US\_09** | Confirm payments | Passengers shall confirm their tickets payment to proceed. |
| **US\_10** | Print tickets | Print their ticket list. |
| **US\_11** | Add a new passenger user | Employee and admin creates an account for a new passenger |
| **US\_12** | Add a new employee user | Admin creates an account for a new employee |
| **US\_13** | Manage employee | Admin can update, delete or view employees |
| **US\_14** | Passengers’ list | View all the passengers |
| **US\_15** | Accounting | Admin can view the overall ticket payment |
| **US\_16** | Search a passenger | Search for a specific passenger |
| **US\_17** | Search an employee | Admin searches for a specific employee |
| **US\_18** | Update passenger | Admin and employee updates personal information of a specific passenger. |
| **US\_19** | Update employee | Admin updates personal information of a specific employee |
| **US\_20** | Delete user | Admin can delete an account of an existing passenger or employee |
| **US\_21** | Approve ticket | Admin and employee approves tickets for the passengers |
| **US\_22** | View profile | Passenger, employee and admin can view their profiles |

|  |  |  |
| --- | --- | --- |
| **US\_23** | View passenger’s ticket | Admin and employee can view all tickets |
| **US\_24** | Manage tickets | Admin and employee can manage tickets , updating or deleting them. |
| **US\_25** | User logs out | Admin, employee and passenger log out from their accounts |

**4.1.1.2 User Scenarios Extended**

1. **US\_01** – User logs in

1. User chooses his user type: passenger/admin/employee
2. User is redirected to the log in page
3. User enters his username and password
4. User presses the log in button
5. If data is correct the user is redirected to his profile page
6. If data is not correct an error message will be shown and user repeats the  process from step b.

2. **US\_02** – Change password

1. User logs in following the steps in **US\_01.**
2. User chooses the menu “Change Password”
3. User types his old password and his new password (2 times)
4. User presses the button “Save Changes”
5. If the old password is correct and if the new password is the same in both fields  the user is alerted: “Password was changed successfully!”
6. In case the old password is wrong or the new password is not the same in both  fields then user is alerted: “Password was not changed. Please try again.”

3. **US\_03** – Change the profile avatar

1. User logs in following the steps **US\_01**
2. User clicks on “Profile Avatar”
3. User clicks on “Choose file” to choose a local file as a profile avatar
4. After users picks the file , he/she clicks “Update Profile”
5. A pop-up is shown “Profile changed successfully”

4. **US\_04** – Update their information

1. User logs in following the steps **US\_01**
2. User clicks on “Update”
3. User fills the parts that want to change
4. After users picks the file, he/she clicks “Update Profile”
5. A pop-up is shown “Your profile has been updated”

5. **US\_05** – Passengers can view trains

1. Passengers logs in following the steps **US\_01**
2. Passengers click on the “All available Trains” section
3. Now they see a list of all the trains available in the system
   * 1. They search a train by clicking in “Search Train” section
     2. They see all the trains and have the option to search and sort trains

6. **US\_06** – Passenger books a train

1. Passengers logs in following the steps **US\_01**
2. Passengers click on the “Reserve Train” section
3. They are shown the list of trains available
4. Passengers click “Book” to book a train
5. Passenger see the characteristics of that train
6. Passengers click on “Book Train” to proceed
7. A pop-up is shown “Success!” ,”Reserved train , please proceed to check out!”
8. Passengers click “ok” to close the window

7. **US\_07** – Passenger cancel the booking

1. Passengers logs in following the steps **US\_01**
2. Passengers click on the “Cancel Train” section
3. They are shown the list of trains booked
4. Passengers click “Cancel” to cancel a train
5. Passengers are sent to the train characteristics
6. Passengers click the button “Cancel Train” to finish cancellation
7. Pop-up is shows “Success” , “Reserved Train Canceled”

8. **US\_08** – Checkout tickets

1. Passengers logs in following the steps **US\_01**
2. Passengers click on the “Checkout” section
3. They are shown the list of trains to checkout
4. Passengers click “Checkout” to book a train
5. Passenger see the characteristics of that train
6. They add the payment code
7. Passengers click on “Checkout Train” to proceed
8. A pop-up is shown “Success!”
9. Passengers click “ok” to close the window

9. **US\_09** – Confirm payment

1. Passengers logs in following the steps **US\_01**
2. Passengers click on the “Confirm Payments” section
3. They are shown the list of trains to confirm the payment
4. Passengers click “Confirm payment” to confirm the payment
5. Passenger see the characteristics of that train
6. Passengers click on “Confirm payment” to proceed
7. A pop-up is shown “Success!”
8. Passengers click “ok” to close the window

10. **US\_10** – Print ticket

1. Passengers logs in following the steps **US\_01**
2. Passengers click on the “Print” section
3. They are shown the list of trains to print with the system logo
4. Passengers can print their ticket by clicking “Print”
5. The usual printing procedure is set-up.

11. **US\_11** – Add a new passenger user

1. Admin/Employee logs in following the steps **US\_01**
2. Admin/Employee clicks on “Add a New Passenger”
3. Admin/Employee fills all the empty fields
4. Admin/Employee double checks if the data entered is correct
5. Admin/Employee clicks the button “Add Passenger”
6. Pops-up: “Passengers account added successfully”
7. Admin/Employee clicks the button “ok” and stays in the same page.
   * 1. The data is saved in the database and the  account is created.

12. **US\_12** – Add a new employee user

1. Admin logs in following the steps **US\_01**
2. Admin clicks on “Add a New Passenger”
3. Admin fills all the empty fields
4. Admin double checks if the data entered is correct
5. Admin clicks the button “Add Passenger”
6. Pops-up: “Employee account added successfully”
7. Admin clicks the button “ok” and stays in the same page.
   * 1. The data is saved in the database and the  account is created.

13. **US\_13** – Manage employee

1. Admin logs in following the steps **US\_01**
2. Admin clicks on “Manage Employee” section
3. A list of all the employees is shown
4. Admin can choose to:
   * 1. Update employee info by clicking “Update”
     2. Delete employee by clicking “Delete”
     3. View info by clicking “View”

14. **US\_14** – Passengers’ list

1. Admin/Employee logs in following the steps **US\_01**
2. Admin/Employee clicks on “Manage Passengers” section
3. A list of all the passengers is shown
4. Admin/Employee can choose to:
   * 1. Update employee info by clicking “Update”
     2. Delete employee by clicking “Delete”
     3. View info by clicking “View”

15. **US\_15** – Accounting

1. Admin logs in following the steps **US\_01**
2. Admin clicks on “Accounting” section
3. In the “View : Ticket payment” section
4. Admin is sent to a dashboard where he/she can see the performance.

16. **US\_16** – Search a passenger

1. Admin/Employee logs in following the steps **US\_01**
2. Admin/Employee clicks on “Manage Passengers” section
3. A list of all the passengers is shown
4. There is a search box where u can search for a specific passenger

17. **US\_17** –Search an employee

1. Admin logs in following the steps **US\_01**
2. Admin clicks on “Manage Employee” section
3. A list of all the passengers is shown
4. There is a search box where u can search for a specific employee

18. **US\_18** –Update passenger

1. Admin/Employee logs in following the steps **US\_01**
2. Admin/Employee clicks on “Manage Passengers” section
3. A list of all the passengers is shown
4. Admin/Employee clicks “Update”
5. Admin/Employee double checks the data entered
6. Admin/Employee clicks the button “Update Passenger”
7. Pops-up: “Passengers account updated successfully”
8. Admin/Employee clicks the button “ok” and stays in the same page.

19. **US\_19** – Update employee

1. Admin logs in following the steps **US\_01**
2. Admin clicks on “Manage employee” section
3. A list of all the employee is shown
4. Admin clicks “Update”
5. Admin double checks the data entered
6. Admin clicks the button “Update employee”
7. Pops-up: “Employee account updated successfully”
8. Admin clicks the button “ok” and stays in the same page.

20. **US\_20** – Delete user

1. Admin/Employee logs in following the steps **US\_01**
2. Admin/Employee clicks on “Manage Passengers/Employee” section
3. A list of all the passengers is shown
4. By clicking “Delete”, the wanted user is deleted

21. **US\_21** – Approve ticket

1. Admin/Employee logs in following the steps **US\_01**
2. Admin/Employee clicks on “View : Pending” section
3. A list of all the tickets is shown
4. By clicking “Approve” , user approves the ticket
5. A form with the ticket info is shown
6. Admin/Employee has to select the state of the ticket in the “Confirmation” field
7. Then click “Confirm Payment”

22. **US\_22** – View profile

1. Users logs in following the steps **US\_01**
2. User clicks on “View” section
3. A page with the users personal information is shown

23. **US\_23** – View passenger’s ticket

1. Admin/Employee logs in following the steps **US\_01**
2. Admin/Employee clicks on “View : Approved” section
3. A list of all the tickets is shown
4. Admin/Employee can search a particular ticket
5. **US\_24** –Manage tickets
6. Admin/Employee logs in following the steps **US\_01**
7. Admin/Employee clicks on “Tickets” section
8. Admin/Employee clicks on “Manage” section
9. A list of all the tickets is shown
10. Admin/Employee can search a particular ticket
11. Admin/Employee can click “Manage” or “Delete”
12. **US\_25** – User logs out
13. Users logs in following the steps **US\_01**
14. Users click “Log Out” section
15. Users are sent back to log in page

**Worked by Algis Miza**

Use Cases-Worked by Klea Dushku

**1-** UC\_01 - Log in

|  |  |
| --- | --- |
| Name | User logs in |
| Summary | User enters credentials to access their account |
| Actor | Passenger, employee, administrator |
| Description | User types email and password, after they are verified, they gain access to the account |
| Precondition | User must be previously registered and have an active account. User must choose the role (passenger/employee/admin) before logging in |
| Alternatives | If the credentials are wrong user cannot log in. If the user has forgotten the password, an email will be sent to log in. |
| Post condition | User is logged in on their account |

1. UC\_02 - Change password

|  |  |
| --- | --- |
| Name | Change password |
| Summary | User goes to corresponding page to change  their password |
| Actor | Passenger, employee, administrator |
| Description | User goes to ‘user’s profile’ and clicks on change password. User types old password and new one to change it. |
| Precondition | User should be logged in. The old password should be typed correctly and the fields with the new password should match. (verification  may be done by email) |
| Alternatives | If the fields are not filled correctly, the password cannot be changed. User can try again. |
| Post condition | After pressing ‘change password’ button, old password is replaced with new one. |

1. UC\_03 - Book ticket

|  |  |
| --- | --- |
| Name | Book ticket |
| Summary | Passenger can book a ticket for a train of their choice. |
| Actor | Passenger |
| Description | User clicks on ‘book train’ in menu bar, then clicks on ‘book’. User chooses a train in the list or searches a specific train in the search bar, then they click the ‘book’ button, which redirects to a form that shows the information of the booking. User clicks on ‘book train’ to  confirm. |
| Precondition | User should be logged in. This feature is only for passengers. |
| Alternatives | User can book multiple tickets. |
| Post condition | After pressing ‘book’ button, the user will  have to proceed to checkout (pay) the ticket. |

1. UC\_04 - Checkout ticket

|  |  |
| --- | --- |
| Name | Checkout ticket |
| Summary | Passenger should make the ticket payment in the checkout section. |
| Actor | Passenger |
| Description | User clicks on ‘tickets’ in the menu bar, then on ‘checkout’. User chooses the ticket they want to checkout from the list. User is redirected to a form with the ticket information. User enters the payment code to  verify the payment. |
| Precondition | This feature is for users only. The ticked should be booked first for it to appear in the  checkout list. |
| Alternatives | If the payment code is not written, the booking cannot be done.. |
| Post condition | After clicking ‘checkout’ button, user will have to confirm this action. |
|  |  |
|  |  |

1. UC\_05 - Confirm payment

|  |  |
| --- | --- |
| Name | Confirm payment |

|  |  |
| --- | --- |
| Summary | Passenger should confirm the payment. |
| Actor | Passenger |
| Description | User clicks on ‘tickets’ in the menu bar, then on ‘confirm payment’. User chooses the ticket they want to confirm from the list. User is redirected to a form with the ticket  information. User presses ‘confirm’ button. |
| Precondition | This feature in only available for passengers. The ticket should be checked-out first to  appear in the confirmation list. |
| Alternatives | User can confirm multiple payments. |
| Post condition | After clicking ‘confirm’ button, user will have to wait for approval of the booking. |

6 – UC\_06 - Print ticket

|  |  |
| --- | --- |
| Name | Print ticket |
| Summary | Passenger can print the ticket. |
| Actor | Passenger |
| Description | User clicks on ‘tickets’ in the menu bar, then on ‘print’. User can view the approved ticket and save/print it in pdf format. |
| Precondition | User should be logged in as passenger. |
| Alternatives |  |
| Post condition | After clicking ‘print’ button, user can  save/print the ticket as a pdf. |

1. UC\_07 - Update profile

|  |  |
| --- | --- |
| Name | Update profile |
| Summary | Users can update their profile / add additional information |
| Actor | Passenger, employee, admin |
| Description | User clicks on ‘profile’ in the menu bar, then on ‘update profile’ to change their profile.  User can view their profile and write a bio, change their username, phone number, email, address. |
| Precondition | All users can do this action. |
| Alternatives |  |
| Post condition | After clicking ‘update’ button, the changes are applied to the profile. |

1. UC\_08 - User adds a profile picture

|  |  |
| --- | --- |
| Name | Add profile picture |
| Summary | Users can put a picture in their profile |
| Actor | Passenger, employee, admin |
| Description | User clicks on ‘profile’ in the menu bar, then on ‘profile picture’ to add picture. User can select a picture from their device and confirm  the update. |
| Precondition | All users can do this action. |
| Alternatives |  |
| Post condition | After clicking ‘update’ button, the changes are applied to the profile. |

9 – UC\_09 - Add a train

|  |  |
| --- | --- |
| Name | Add train |
| Summary | Employees and managers can add a train and its information to the system |
| Actor | Employee, admin |
| Description | User clicks on ‘train’, ‘add train’ in the menu bar and adds train. |
| Precondition | User must fill out all necessary information about the train.  User should be logged in as either admin/employee. |
| Alternatives | If any field is not filled out, the train cannot be added. |
| Post condition | After clicking ‘add train’ button, the train is added to the system. |

1. UC\_10 - Manage trains

|  |  |
| --- | --- |
| Name | Manage trains |
| Summary | Employees and managers can change train information and can cancel trains. |
| Actor | Employee, admin |
| Description | User clicks on ‘train’, ‘manage trains’ in the menu bar to change train info. User can click on cancel train to remove a train from the system. User can click on manage train to change train info like name, number,  departure, destination, etc. |

|  |  |
| --- | --- |
| Precondition | User should be logged in as admin/employee |
| Alternatives |  |
| Post condition | After clicking ‘confirm’ button, the changes are applied. |

1. UC\_11 - Add passenger

|  |  |
| --- | --- |
| Name | Add passenger |
| Summary | Employees and managers can create passenger profiles. |
| Actor | Employee, admin |
| Description | User clicks on ‘passengers’, ‘add passengers’ in the menu bar to add passenger. |
| Precondition | User must fill out all the required information to add the passenger.  User should be logged in as either admin/employee. |
| Alternatives | If any required field is missing, the passenger cannot be added. |
| Post condition | After clicking ‘confirm’ button, the passenger profile is created. |

1. UC\_12 - Manage passenger (update/delete/view passenger profile)

|  |  |
| --- | --- |
| Name | Manage passenger |
| Summary | Manager and employee can modify passenger profiles. |
| Actor | Admin, employee |
| Description | User clicks on ‘passengers’, ‘manage passenger’ in the menu bar to edit passenger information (but does not have access to the  password). |
| Precondition | User should be logged in as admin or employee |
| Alternatives | User can also delete passengers and view passenger profiles. |
| Post condition | The passenger information will be updated. |

1. - UC\_13 - Approve ticket

|  |  |
| --- | --- |
| Name | Approve ticket |

|  |  |
| --- | --- |
| Summary | Employees and managers can approve the tickets that have been paid by the passengers. |
| Actor | Employee, admin |
| Description | User clicks on ‘tickets’, ‘approve tickets’ in the menu bar to approve tickets. User clicks on ‘approve ticket’ button to approve the  ticket. |
| Precondition | User should be logged in as either admin/employee |
| Alternatives | User can click on decline to cancel the ticket. |
| Post condition | The ticket will appear in the approved tickets section. (the passenger may be notified for the  ticket decline) |

1. – UC\_14 - Manage ticket

|  |  |
| --- | --- |
| Name | Manage ticket |
| Summary | Employees and managers can change the tickets as necessary. |
| Actor | Employee, admin |
| Description | User clicks on ‘tickets’, ‘manage tickets’ in the menu bar to manage tickets. User clicks  on ‘manage’ button to change the ticket information. |
| Precondition | User should be logged in as either an admin  or employee. |
| Alternatives | User clicks on ‘delete to delete a ticket |
| Post condition | The ticket will be updated or deleted. (the passenger may be notified about these  changes) |

1. – UC\_15 - Add employee

|  |  |
| --- | --- |
| Name | Add employee |
| Summary | Manager can create employee profiles. |
| Actor | Admin |
| Description | User clicks on ‘employees’, ‘add employee’ in the menu bar to add an employee.  User fills out the necessary information to  create the employee profile. |
| Precondition | User should be logged in as an admin. The other roles do not have this feature. |
| Alternatives | If any of the required fields is not filled out, the profile cannot be created. |

|  |  |
| --- | --- |
| Post condition | The employee will be registered in the database, the employee can log in with the credentials provided by the admin |

1. – UC\_16 - Manage employees (update/delete/view profile of employee)

|  |  |
| --- | --- |
| Name | Manage employee |
| Summary | Manager can modify employee profiles. |
| Actor | Admin |
| Description | User clicks on ‘employees’, ‘manage employee’ in the menu bar to edit an employee’s information (but does not have access to the password). |
| Precondition | User should be logged in as admin. |
| Alternatives | User can also delete the employee and view  the employee profile. |
| Post condition | The employee information will be updated. |

17- UC\_17 - Accounting

|  |  |
| --- | --- |
| Name | View finances |
| Summary | Manager can view information about finances. |
| Actor | Admin |
| Description | User clicks on ‘accounting’, ‘view finances’ in the menu bar to view total amount from  ticket payments, number of bookings, ticket status (paid/approved/not paid/not approved) |
| Precondition | User should be logged in as admin. |
| Alternatives |  |
| Post condition | The information is shown. |

1. – UC\_18 - Trains

|  |  |
| --- | --- |
| Name | All available trains |
| Summary | Users can view a list of all available trains |
| Actor | Admin, employee, passenger |
| Description | User clicks on ‘trains’, ‘all available trains’ in the menu bar to view a list of all the registered trains and their respective information. |
| Precondition | All logged in users can view this info. |
| Alternatives |  |
| Post condition | The list is shown to the user. |

1. – UC\_19 - Search train

|  |  |
| --- | --- |
| Name | Search train |
| Summary | Users can search a train by anyy info related to the train. |
| Actor | Admin, employee, passenger |
| Description | User clicks on ‘trains’, ‘search train’ in the menu bar, the types the information in the search bar (it can be any information related to the train, like fee, departure time,  destination, etc.) |
| Precondition | Any information can be used. It is not case- sensitive. |
| Alternatives | If no information can be found that corresponds to the searched info, the list will be empty. |
| Post condition | The list of trains with the related info is shown to the user. |

1. – UC\_20 – Search passenger /View passenger list

|  |  |
| --- | --- |
| Name | Search passenger |
| Summary | Admin and employee can search a specific passenger |
| Actor | Admin, employee |
| Description | User clicks on ‘passengers’, ‘manage passengers’, in the menu bar to view a list of all the registered passengers and their  respective information. To search the user can type any information regarding the passenger. |
| Precondition | User should be logged in as admin or  employee, passengers do not have this feature |
| Alternatives | If there are no passengers matching the info typed in the search box, the list will be empty |
| Post condition | The passenger or list of passengers with the related info is shown to the user. |

1. – UC\_21 – Search employee /View employee list

|  |  |
| --- | --- |
| Name | Search a specific employee |
| Summary | Admin can search a specific employee |
| Actor | Admin |
| Description | User clicks on ‘employees’, ‘manage employees’ in the menu bar to view a list of |

|  |  |
| --- | --- |
|  | all the registered employees and their respective information. To search the user can type any information regarding the employee. |
| Precondition | If there are no employees matching the info  typed in the search box, the list will be empty |
| Alternatives | If there are no passengers matching the info typed in the search box, the list will be empty |
| Post condition | The employee or list of employees with the related info is shown to the user. |

1. – UC\_22 – Log out

|  |  |
| --- | --- |
| Name | User logs out |
| Summary | User has finished their activity |
| Actor | Admin, employee, passenger |
| Description | When the user has finished interacting with the web application, they can log out by clicking on ‘log out’ in the menu bar. |
| Precondition | User must be logged in first |
| Alternatives | User can log out at any time |
| Post condition | The user has logged out and is no longer using their account. |

**4.2 Behavioral Diagrams**

**4.2.1 Use Case Diagrams**

**A diagram of a process

Description automatically generated**

BASIC OPERATIONS

A diagram of a company

Description automatically generated

Relationship between Employee and Passanger

A diagram of a workflow

Description automatically generated

Relationship between Admin and Employee

A diagram of a user interface

Description automatically generated

Relationship Admin and Pasanger

A diagram of a company

Description automatically generated

Booking Ticket