|  |  |
| --- | --- |
| **Project title** | **Train Booker** |
| **Author(s)** | **Itu Anca, Mihuț Alina** |
| **Group** | **30422** |

# Task Description

Our project is a Java application used for booking and buying train tickets.

The **user** class will have the following attributes: first name, last name, email, password phone number and a residence place;

The **trip** class: train number, departure time, arrival time, trip length, departure station, arrival station, trip length in minutes;

The **train** class: train number, departure station, stations, time between stations, seats for 1st class, seats for 2nd class, seats for 1st sleeping class, seats for 2nd sleeping class;

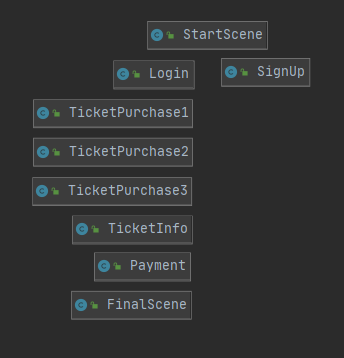
The user is able to:

* **Sign up** if he doesn’t have an account:
* Introduce the first name, last name, email, phone number, residence place and a password
* The user will have to introduce data in all the text fields; the phone number and the email must be valid; the password must have at least 6 characters; the user cannot create an account using an email which already belongs to another account;
* **Log in** into an already existing account with the email and password
* Once logged in, the user is able to search for train tickets:
* He selects:
* the departure station and the destination
* a date in the following 7 days
* Then, the train options with the specified attributes will be displayed, along with travel details:
  + Train number
  + Departure & arrival time
  + Departure & arrival stations
  + Length of the trip
* The user selects one of the options and he goes to the next page, where he selects:
  + The number of tickets he wants to purchase
  + The travel class
  + The status of the travelers (regular adult, child, pupil, student, retired person)
* The trip details (selected train number, departure date, departure & arrival station, departure & arrival time, trip length, the number of tickets, the status of the travelers and the class), along with the price per ticket and the total price will be displayed
* In the next step the user has to make the payment, so he introduces his card details, which have to be valid
* He confirms the purchase and the booking is recorded in the system.

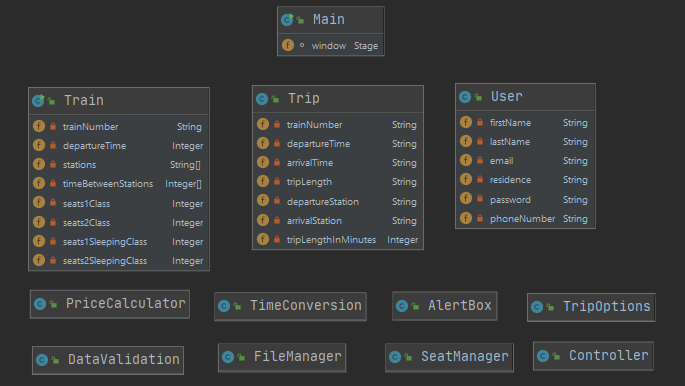
**Functionalities:**

* **Seat manager**: each travel class in each train for each date has a standard number of seats which updates after each purchase. So, if the user tries to buy one or more tickets for a specific train and class where there are not enough seats available, an alert message will appear and the user will have to change the specifications made regarding the trip.
* **Price calculator**: the unit price is decided depending on the class and traveler status. For example, children younger than 6 years old travel for free, students and retired people travel for free and the pupils have a discount at the second class.

# Class Diagram



In the first diagram one can see the flow of the scenes, each one being implemented in a particular class. The first scene is the StartScene where the user can choose to go to the Login page or to SignUp. In the SignUp page, after he creates an account, he is redirected to the Login page. After he logs in, he goes through a sequence of pages where he has to choose the details of the trip. In the TicketInfo page he can see all the information regarding his trip and then he is sent to the page where he has to introduce the details of the payment. In the FinalScene his purchase is confirmed and he is redirected to the StartScene.



In the diagram above there are the object classes (Train, Trip and User) with their attributes and the classes containing the functionalities.