The Design Document - DriveUp

Irfan Durić
Dino Šaćiragić
Haris Kovačević
Ahmed Nurović
Harun Šaranović

1. Document Revision History

Revision 1.0 March 26, 2019 – initial version: In initial version we made several things in our web application. Firstly we made home page and added some basic features and explanation on how application works and what it gives. We then added options for login and contact. In login we have 2 types of login. One is for drivers and one is for customers. Drivers need to write more informations about them(username, e-mail, phone number) while customers who are looking for driver need only username and e-mail. In contact there are informations on how can users get in touch with us if they need any help.

Revision 1.1 April 2, 2019 – In this revision we added additional requirements. Here we added options when users(drivers) register and login they can add more informations about them(type of car, how many seats, where they can drive, when they are able to drive, pictures of car). Also there are categories where customers can search for specific drivers. There are options like (in which city they can drive, price, how many seats). We also added options that customers send message to drivers, and also they can see their profile where they can see other info like phone number or e-mail that can be used to contact them. We didn't remove anything yet, but if we find some mistake or some better idea or any kind of improvement we will remove things and add new better ones.

2. Overview of the application

The DriveUp web application is an application which will be used by drivers and potential customers which want to go from point A to point B.
This application tends to be a middleman application which allows the communication between the driver and the customer. It will also have an registration where drivers and customers will have seperate forms to fill out in order to make the account, drivers will be asked to pay the monthly subscription for using the application, on the other hand the customer account is free from any payment towards the application and its use.
This application will also feature a list of those same drivers which will show their details, those details are going to be : Type of car, Availability, Working days, Number of seats, Description, Contacts options and price. This feature will make it easy for customers to choose their driver based on their most important information and which suits them the best. As previously mentioned there will be communication between the drivers and customers on the application, this chat option is for making deals and setting up meetings with your driver.
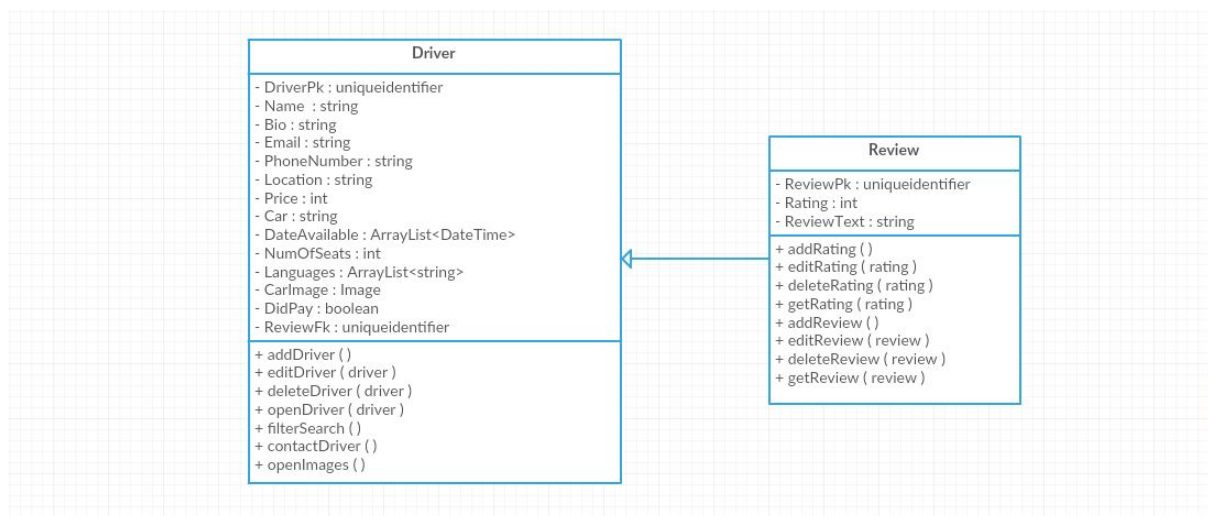
When the customer finally chooses a driver and pays him for his service, the customer has the option to review the driver that the choose on his profile, this will be important for the driver since he needs to prove customers that he is the best driver, which means that the best driver gets the most jobs like almost every freelance application.

Further in the document these features and details will be explained in detail and specification of the application.

3. System Decomposition/Design
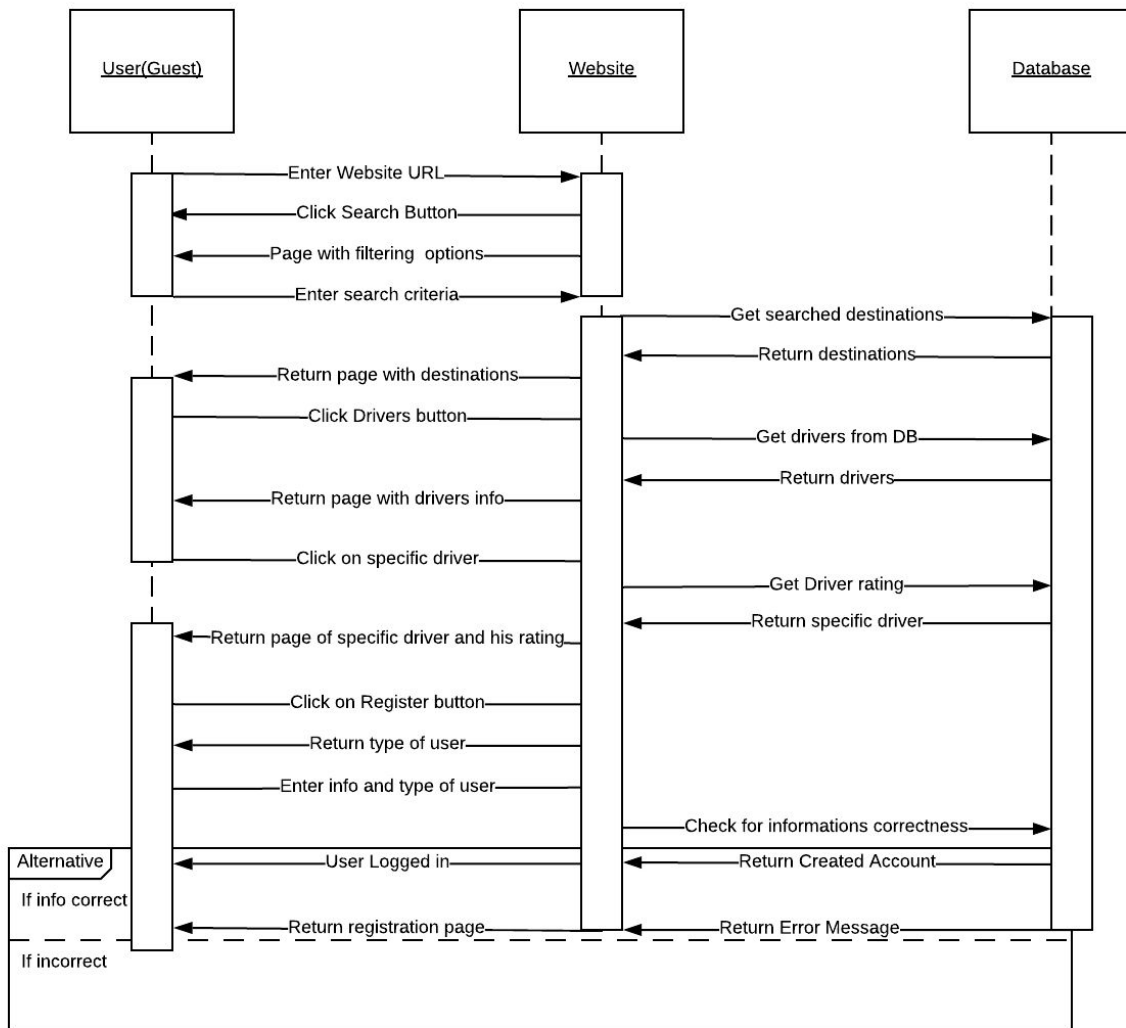
Modules
- **AppModule -** Module where all other modules are imported
- **FormsModule -** Module used for creating forms with Angular
- **DriverModule -** Module of the class Driver
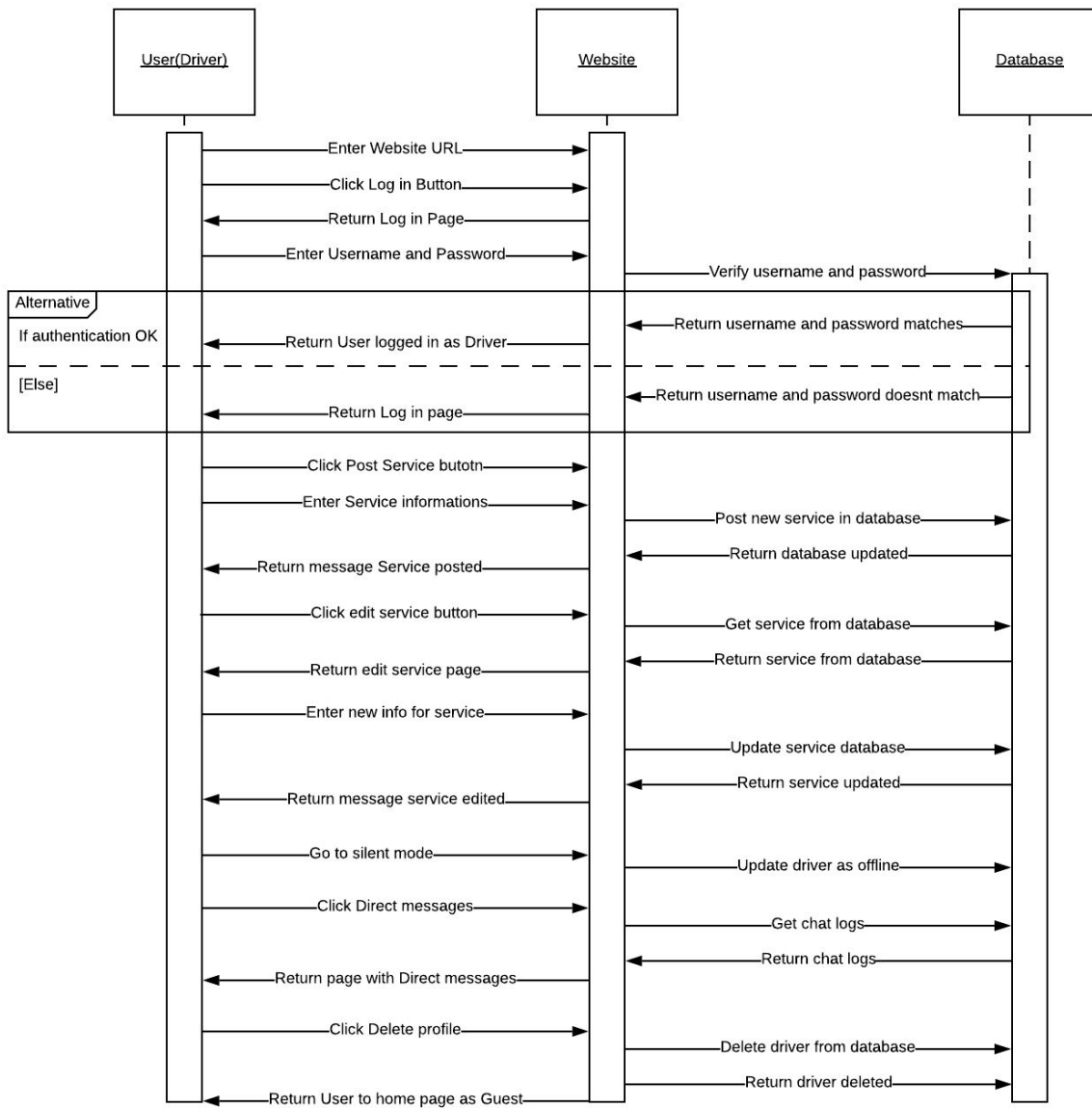- **ReviewModule -** Module of the class Review
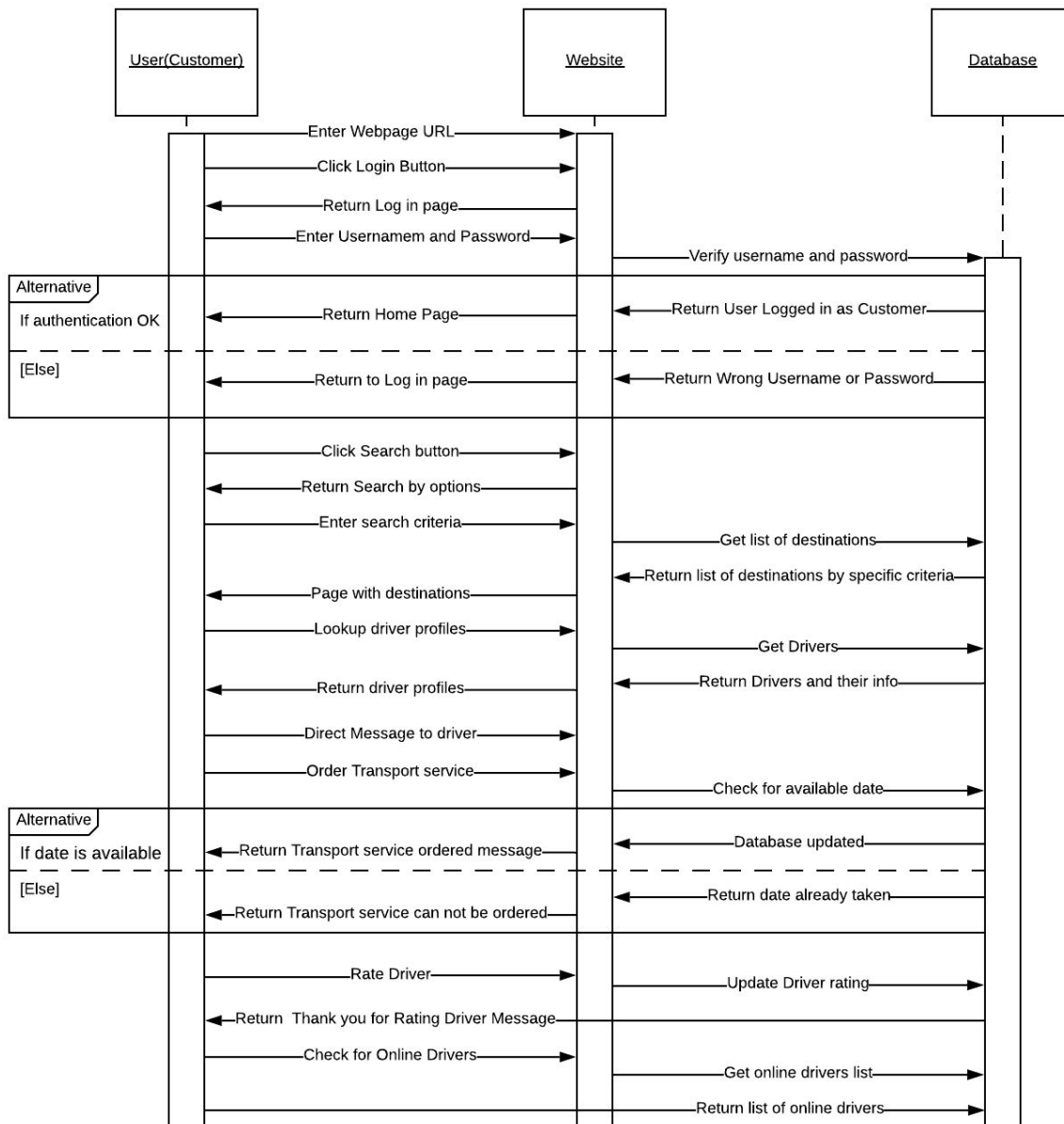
4. Dynamic Models: Sequence Diagrams

4.1 Guest (User did not register)

```
  User(Guest)              Website                 Database

  ──Enter Website URL──────►
  ◄──Click Search Button───
  ◄──Page with filtering options──
  ──Enter search criteria──►
                            ──Get searched destinations──►
                            ◄──Return destinations──
  ◄──Return page with destinations──
  ──Click Drivers button──
                            ──Get drivers from DB──►
                            ◄──Return drivers──
  ◄──Return page with drivers info──
  ──Click on specific driver──
                            ──Get Driver rating──►
                            ◄──Return specific driver──
  ◄──Return page of specific driver and his rating──
  ──Click on Register button──
  ◄──Return type of user──
  ──Enter info and type of user──
                            ──Check for informations correctness──►
Alternative
If info correct
  ◄──User Logged in──       ◄──Return Created Account──
If incorrect
  ◄──Return registration page──   ◄──Return Error Message──
```

4.2 Driver Sequence Diagram

User(Driver)	Website	Database

Enter Website URL

Click Log in Button

Return Log in Page

Enter Username and Password

Verify username and password

Alternative

If authentication OK

Return User logged in as Driver

Return username and password matches

[Else]

Return username and password doesnt match

Return Log in page

Click Post Service butotn

Enter Service informations

Post new service in database

Return message Service posted

Return database updated

Click edit service button

Get service from database

Return edit service page

Return service from database

Enter new info for service

Update service database

Return message service edited

Return service updated

Go to silent mode

Update driver as offline

Click Direct messages

Get chat logs

Return page with Direct messages

Return chat logs

Click Delete profile

Delete driver from database

Return User to home page as Guest

Return driver deleted

4.3 Customer Sequence Diagram

Sequence diagram with lifelines: User(Customer), Website, Database.

- Enter Webpage URL → Website
- Click Login Button → Website
- Return Log in page ← Website
- Enter Usernamem and Password → Website
- Verify username and password → Database

Alternative — If authentication OK:
- Return Home Page ← Website
- Return User Logged in as Customer ← Database

[Else]:
- Return to Log in page ← Website
- Return Wrong Username or Password ← Database

- Click Search button → Website
- Return Search by options ← Website
- Enter search criteria → Website
- Get list of destinations → Database
- Return list of destinations by specific criteria ← Database
- Page with destinations ← Website
- Lookup driver profiles → Website
- Get Drivers → Database
- Return driver profiles ← Website
- Return Drivers and their info ← Database
- Direct Message to driver → Website
- Order Transport service → Website
- Check for available date → Database

Alternative — If date is available:
- Return Transport service ordered message ← Website
- Database updated ← Database

[Else]:
- Return Transport service can not be ordered ← Website
- Return date already taken ← Database

- Rate Driver → Website
- Update Driver rating → Database
- Return Thank you for Rating Driver Message ← Website
- Check for Online Drivers → Website
- Get online drivers list → Database
- Return list of online drivers ← Database

5. User Interface

5.1. Registration Form

- User should enter his data into forms
- We decided to provide lazy registration form
- All fields in the form are required
- Once data is entered, user should press the **Register Button**
- User can choose between two account types **(Driver and Customer)**

5.2. Login Form

- User log into the application using his email and password
- Both **Email** and **Password** fields are required
- Once data is entered, user should press the **Login Button** to log in
- If user do not have account, he can click **Register Here** link and he will be provided with **Registration Form**

5.3. Drivers and Transportation Services list

- Customer who seeks transport services can look for available drivers
- He can see **information about driver** and the **price**
- By clicking the **Contact Button**, application will provide the messages form to the customer

5.3. Drivers and Transportation Services list
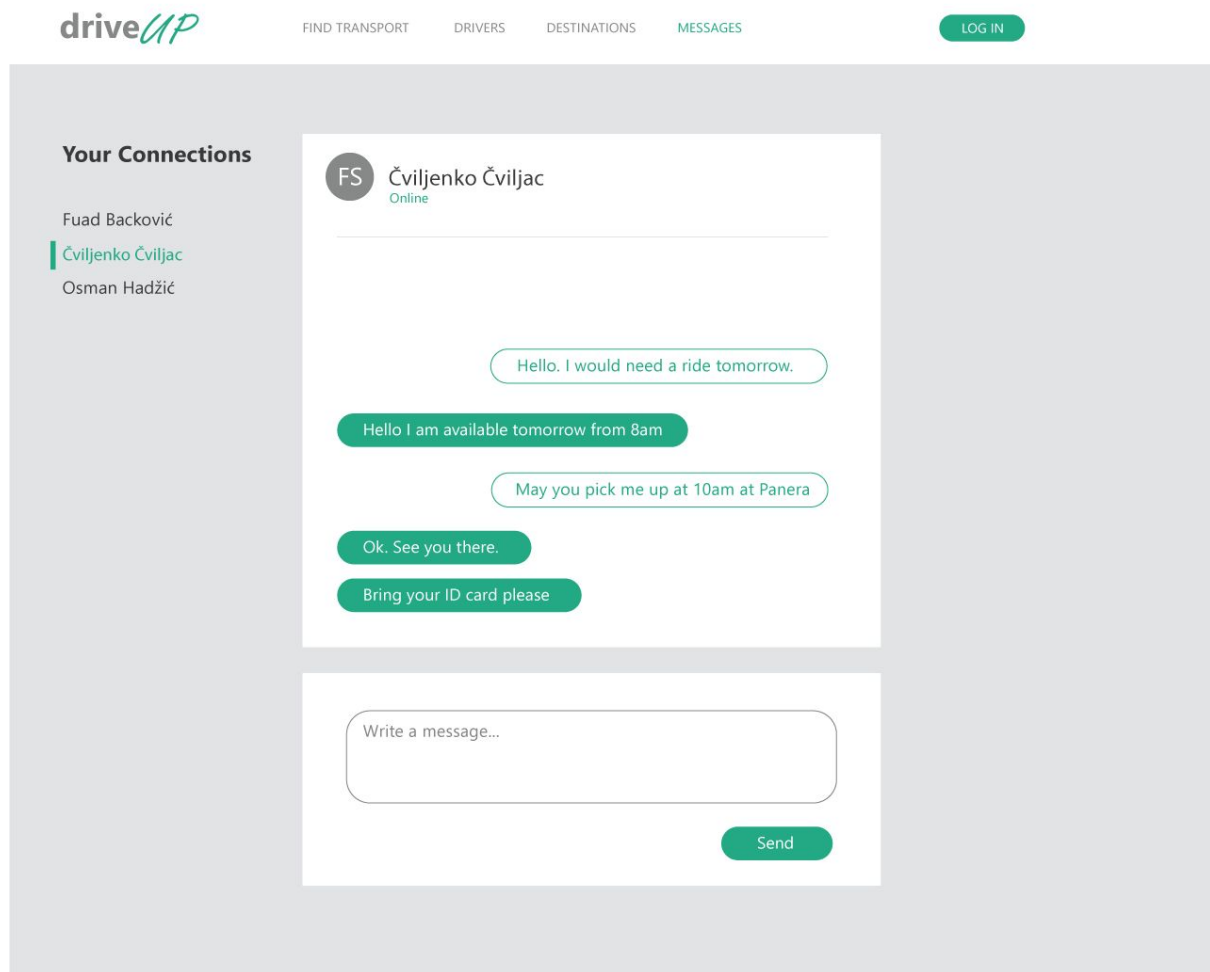
- **Upper window** shows all the **messages** driver and customer have exchanged
- Right sided messages are messages from user who looks at the form
- For writing **new message**, user uses form in the **Bottom window**
- Once user finishes a message, he sends it by pressing **Enter** or clicking the **Send button**
- At the left side is the list consisted of all users whom with the **user had been in contact**