



Menoufia University
Faculty of Computers and Information

RAKNA GUIDE

2019

smart Parking system
S P S

S P S

Production Name *Rakna Guide*



Team Members

Full Name	Department
Abdalrahman Mohamed Omran	CS
Azer Nasr Saad Botrous	CS
Islam Yousry Tawfik Qelash	CS
Hazem Samy Allam	CS
Saeed Elsayed Eldeed	CS
Hassan Ragab Elkhadrawy	CS

Acknowledgement

We would like to express our appreciation to our supervisor, Dr. Osama Abdel Raouf who has cheerfully answered our queries, provided us with materials, checked our examples, assisted us in a myriad way with the writing & helpfully commented on earlier drafts of this project. We also very grateful to our demonstrator Mohamed Mostafa, our friends, family for their good humor & support throughout the production of this project.

Smart Parking System

Table of Contents

1. Abstract.....	4
2. Tools used in the project.....	6
3. Challenges that may represent a problem or a shift in the project.....	6
4. External interface requirements	18
5. Hardware interface	28
6. Software interface & communication interface	30
7. System features	31
8. Functional & Non-functional requirements	39
9. Software quality attributes	46
10. Software importance	47
11. System Modeling	48
12. Clients-server architecture	57
13. Test cases	58
14. Databases	61
15. Architectural pattern	62
16. Development and implementation	63
17. Future plans	83

➤ Abstract

Rakna Guide is a platform that can be accessed via mobile app, car owner rent a space for his car from a garage owner after checking the nearest garages and get the suitable one for him, having all garage data and access the security situation of the car via camera.

● Car owner application review: -

- On opening the application for the first time, registration screen appears then car owner enters all required info about his car such as his name, id number, Car license, Visa number(optional) number and his phone number and a photo of his car.
- The App will ask for the GPS permission so that app can use for make its services.
- He has two options to make: first one is to search for the nearest garages around his existence place, second one is choosing destination of his ride to also search for available garages.
- When choosing the garage, all info about that garage (Photos, Area, Height) and his owner appears.

Smart Parking System

- When choosing his suitable one, App leads the owner to the garage destination by drawing the route on map.
- On arriving, he had to confirm that and from that time it's being calculated until car owner come and take his car.
- There are two methods for payment Cash or via smart cards.
- A needed screen appears for rating the garage and its services.

- **Garage owner application review:** -

- On opening the application for the first time, registration screen appears then garage owner enters all required info about his garage such as his name, id number, garage (Photos, Area, Height, Location), Visa number(optional) number and his phone number.
- When car owner choosing his garage, app notify the garage owner of his garage rent operation.
- There are two methods for payment Cash or via smart cards the garage owner should determine the payment method.
- A needed screen appears for rating the car owner and his treat.

- The used tools in this project are
Android , PHP , MySQL DBMS , RTSP protocol , Volley Server-Client API , Firebase , Mapbox , Braintree , Retrofit restful API.

- What Challenges that may represent a problem or a shift in the project?

First there are multiple problems in many directions some of which are independent while others are dependent on One another.

■ Starting Off from the Network:

losing connection to the internet will affect the following:

- how to access a black box camera and get live streams from it.
- how to save user actions when losing internet connections during leaving the garage?

- how to cancel reservation while losing connection to internet.

■ Next to the User Authority &Management:

- how to manage user profile if car owner sell car?
- If one wanted to take a car he did not own and this car is already registered in the application, He cannot book a garage with this car.

● Answering the Network Challenges

- Each camera has a default link transmit streams to RTSP Server The App uses advanced P2P network transmission technology, remote monitoring using RTSP protocol.
- Data is saved until the Internet is connected again via a local database.
- garage owner has ability to confirm the end of reservation.

- Answering the User Authority & Management:
 - **The new buyer has two option:**
 - ask the x-owner to delete the account.
 - contact with app technical support to make it for him.

➤ Introduction

Using the IEEE SRS standard in the analysis phase of building the software.

1) Purpose

- This SRS document presents a detailed description of the S P S system, version 1.0. It represents the client requirements analysis that defines the functional and non-functional requirements of the Rakna Guide application and its different functionalities. It defines the abilities, reactions from guidelines and limitations of the system. This document will be complete in its scope of the system and the functions required. The system provides a solution to allow garage owner share their empty spaces to another car owner
- Allow the car owner to find a place for their car quickly and safely

2) Document Conventions

The document follows the IEEE format standard (IEEE Std. 830 – 1998).

3) Project Scope

- Rakna Guide is a peer-to-peer garage sharing, Its platforms can be accessed via its websites and mobile apps, car owner rent a space for his car from a garage owner after checking the nearest garages and get the suitable one for him, having all garage owner data and access to his car situation via camera.
- The application is compatible on android platform, it's divided into three parts. 2 of them are android apps, a car owner app version and a garage owner app, the latest part is the server which has all logical operation (payment operation – camera analysis -etc.) and Database.

➤ OVERALL DESCRIPTION

■ Product Perspective

This project represents version one of the S P S management system. All requirements listed herein describe a self-contained system. This project will allow lecturer to share screen & make quizzes to students, calculate their degree automatically by the system and taking the absence.

■ Reason

- System have a lot of problems such the car owner can't know what garages have an empty space to use.
- It wastes time and effort of car owner to get a nearby garage with empty space.
- There are no safety guarantees if you parked the car under the bridge or with Al-Sayes and Al-Sayes would take the car keys to make car maneuvers.

Smart Parking System

- Car owner has no ability to follow his car situation when he is away.
- Lack of central management can be referred to when a problem occurs.
- There is no data about car owner nor garage owner that we can get back to.

■ Goals

Our new system is aimed at a solution that satisfies all parties:

- Using maps to get nearby garages that have empty spaces.
- Always there is safety methods like cameras and car would be on probation.
- Central management represented in company that have all responsibility of its users.

Smart Parking System

- Every car and garage owner would be had* his database in company center so it can be used to solve customers' problems.

➤ Operating Environment

- OS “Windows” , “Linux”
- NetBeans IDE,
- android studio
- Xampp

➤ Product Features

We can subdivide the project into 9 main features,
Details of each of the following functions:

1) Registration

- **Description:** registration screen appears then car owner enters all required info about his car such as:

required data

- his name
- email
- phone number
- password

required data after

- Car license
- Visa number
- photo of his car
- driver license

Smart Parking System

- **Rationale:** It enables new client to check our application features

2) Login

- **Description:** This function allowing a registered car owner to login by his/her email and password. If he's already signed up, he'll find his/her account in database then he can Reserve the garage and he will have All the advantages of the application
- **Rationale:** This provides security to the system by authenticating each client & provides confidence to the user that his/her personal information is secure.

3) Map

- **Description:** by this function enables car owner to find the nearest garage and draw the route to a selected garage.
- **Rationale:** this function gives bunch of information about nearest garages.

4) Notification

- **Description:** this function allowing to track Changes in the garage.
- **Rationale:** this enable to security for the car of being stole

5) History

- **Description:** By this function the car and the garage owner have the whole record of his reserved operation
- **Rationale:** this ensure and enable the two clients having all information about reservation before.

6) Chat

- **Description:** by this function the car owner & the garage owner can communicate with each other through messages on a chat where the car owner can ask the garage owner for anything & the same for the garage owner.

- **Rationale:** this function gives elasticity between the car owner & the garage owner during the reservation.

7) Camera

- **Description:** this function is show automatically live stream by the system when car owner reserve garage that he can use check his car.
- **Rationale:** it is basically the main security function for our system.

8) Payment

- **Description:** Enables the car owner to pay for his usage of an empty garage and garage owner receive a confirmation message of this payment.
- **Rationale:** an easy way for garage owner to earn money and easy payment method for car owner (win win situation).

9) Logout

- **Description:** the car owner can log out of the system himself optionally but before that he is prompted to take his car out of the garage, also the garage owner can log out & in this case everyone is out of the system.
- **Rationale:** the users end everything & logout or being forced to logout by the system.

➤ External Interface Requirements

- **Product Functions:** -
 - Registration
 - Login
 - Map
 - Notification
 - History
 - Chat
 - Camera

- Payment
 - Logout
-
- System users (stakeholders)
 - Car owners
 - Banks.
 - Garage owners.
 - Rakna Guide (Server).

➤ User Interface: -

In the Car Owner App

- Login Screen:

The first scene in the whole system that appears to the user in which the **login** function allowing a registered user to login by his/her email and password or login using third party provider such as Google – Facebook account.

Smart Parking System

If he's already signed up, he'll find his/her app first screen start with map.

- **Map Screen:**

In the map activity user will find his profile image, name and rate in the header of the screen.

The map screen detects his/her current location and find nearest garage around him/her or search to find a garage in another destination he/she want to go.

When the user selects his targeted garage, a small card would appear in the foot part of the screen that holds all information about selected garage.

- **Garage Info Screen:**

The user will find: -

- Garage name
- Garage Images

- Garage rate
- Garage location

There would be several buttons: -

■ Reserve Button

Draw the best route to a garage with the voice telling a user direction.

■ Call Button

It gets the contact number of the garage owner and call him.

■ Chat Button

Create connection between car owner and garage owner using messaging.

■ Share location Button

Send garage location to anyone To facilitate finding way to the garage.

- **Arrive Button**

Determine the actual car owner arriving to a garage and stop map from tracking the car.

- **QR Code & NFC Screen**

Determine that is the car owner who actually reserve this garage and start

timer to count the amount of time car in stayed garage.

- **Bill Screen:**

In this screen the users would have all time information and the cost of this time he has used. In the bottom of the screen there is a required rate bar.

- **Payment Screen:**

In this screen the users can put his visa card number if he hasn't type it before.

- **Main Stage:**

Contains some tabs like: -

- **Rakna Guide Screen Tab:** the function of this tab is to display all information about a garage that car owner parking its car and display a live timer that count the amount of time of parking.
- **Notification Tab:** the function of this tab to make user always connecting with live events that performed/occurring at the garage such as:
 - A Warning Message/Notification telling user security issues about his/her vehicle.

- A notification telling user he/she has a message.
- **Profile Tab:** the function of this tab to display all information about a user and provide ability to make changes to user information such as reset password edit/update photo, car license, car model, driver license, user name and phone number.
 - Logout Menu Item:** that the user can log out of the system himself optionally.
 - History Menu Item:** View all parked user garages in order to quickly select one of them.
- **Chat Tab:** the function of this tab to provide a user the ability to connecting with garage owner via messaging.

- **Camera Tap:** the function of this tab to display live stream of a user car to the owner to strengthen the security issues.

In the Garage Owner App

- **Login Screen:**

The first scene in the whole system that appears to the user in which the **login** function allowing a registered user to login by his/her email and password or login using third party provider such as Google – Facebook account.

If he's already signed up, he'll find his/her app first screen start with Main Stage.

- **Main Stage:**

Contains some tabs like: -

Smart Parking System

- Online rent Screen Tab: the function of this tab is to display all customers whose cars are already in the garage.
- Notification Tab: the function of this tab to make user always connecting with live events that performed/occurring at the garage such as:
 - A Warning Message /Notification telling user
 - security issues about client vehicle.
 - A notification telling user that he has message from his client.
 - A notification telling user that a car owner requested to reserve his garage.

- **Profile Tab:** the function of this tab to display all information about a user and provide ability to make changes to user information such as reset password edit/update photo, national id number, location, garage area, user name and phone number.
- **Logout Menu Item:** that the user can log out of the system himself optionally.
- **History Menu Item:** View all owners of parked cars in their garage to make it easy to choose one of them quickly.
- **Chat Tab:** the function of this tab to provide a user the ability to connecting with garage owner via messaging.
- **Camera Tab:** the function of this tab to display live stream of a user car to the owner to strengthen the security issues

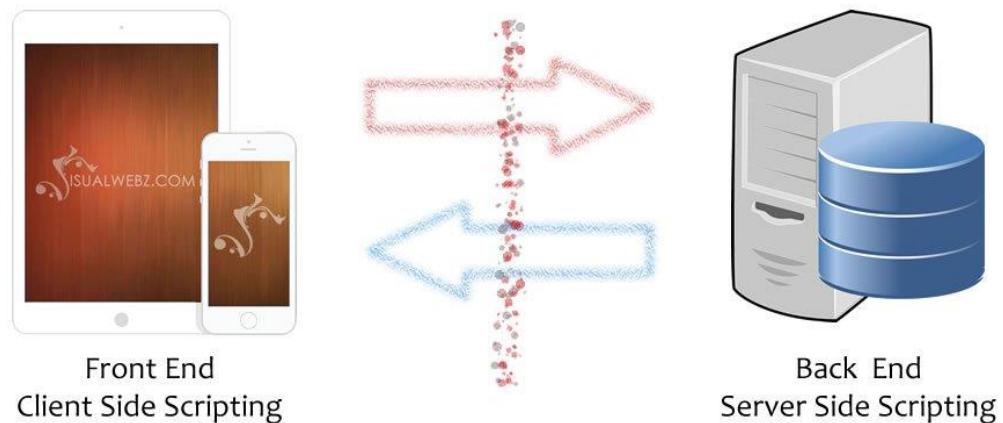
➤ Hardware Interface: -

- IP Camera

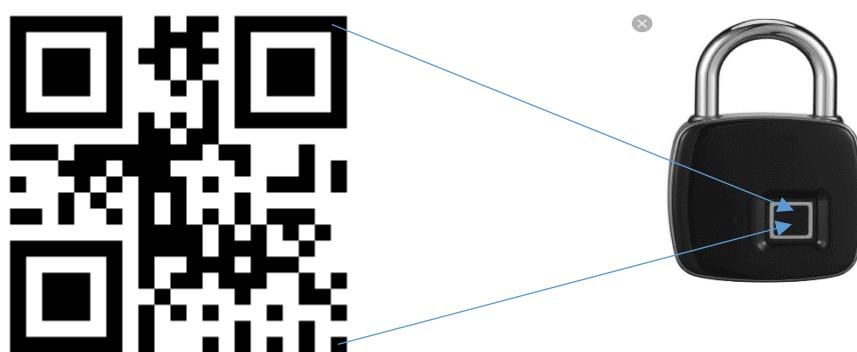


Smart Parking System

■ Backend Server



■ Electronic Locks



➤ Software Interface: -

- Operating System “Windows”, “Linux”
- JDK “Java Development Kit”
- Gradle package
- SQL package
- PHP MySQL
- Client-Server Back-End pattern
- Network Wireless Availability

➤ Communication Interface: -

- Email & Password requiring
- RTSP for transferring streams

➤ System Features

1) Login

Description and Priority: This function allows a registered user to login his account using his frequent email and password. If a user is not registered, the Program should allow the user to enroll first. The system will check both the email and password, when a user attempts to login.

Inputs: Email and Password.

Source: All inputs are provided by user.

Outputs: Indication that user is logged in to the system.

Destination: The outputs are displayed on the screen as well as stored in the system.
Requires: The user provides login information including Email and password.

Pre-Conditions: User is not logged in to system. User has previously enrolled in system.

Post-Conditions: User is logged in to system, OR user is not logged in because he/she entered unrecognized.

2) Map:

Description and Priority: The map screen detect his/her current location and find nearest garage around him/her or search to find a garage in another destination he/she want to go.

Inputs: latitude, longitude.

Source: All inputs are provided by the GPS system.

Outputs: Display garage location on the map.

Destination: users' smart phones.

Requires: the user must open GPS.

Pre-Conditions: the system shared the garage location on the map.

Post-Conditions: the users reserve garage.

3) Garage location:

Description and Priority: Draw the best route to a garage with the voice telling a user direction.

Inputs: source latng, destination latlng.

Source: All inputs are provided by user selection.

Outputs: Best Route.

Destination: users' smart phones.

Requires: the user must open GPS and online Network.

Pre-Conditions: the system shared the best route on the map.

Post-Conditions: the user reach the destination.

4) Chat:

Description and Priority: This feature the car owner & the garage owner can communicate with each other through messages on a chat where the car owner can ask the garage owner for anything & the same for the garage owner. gives elasticity between the car owner & the garage owner during the reservation.

Inputs: message.

Source: All inputs are added by the user.

Outputs: message content & date.

Destination: The outputs are displayed on the screen.

Requires: users logged with system.

Pre-Conditions: an empty message is created.

Post-Conditions: Messages are saved in our database.

5) Electronic lock:

Description and Priority: this function allow the doctor to scan QRcode or use NFC technology to unlock the garage.

Inputs: scan.

Source: All inputs from lock.

Outputs: unlock.

Destination: several system process (online rent).

Requires: online network, support NFC.

Pre-Conditions: server generate QRcode to compare with scanned it from the lock.

Post-Conditions: garage is open.

6) Notification

Description and Priority: This feature *make user always connecting with live events that performed/occurring at the garage.*

Inputs: camera warning and client.

Source: All inputs from camera and client.

Outputs: Notification messages and beeps.

Destination: users.

Requires: online network.

Pre-Conditions: camera detection is available.

Post-Conditions: depending on user action.

7) Camera:

Description and Priority: This feature *display live stream of a user car to the owner to strengthen the security issues.*

Inputs: A garage environment.

Source: All inputs from camera.

Outputs: live streaming and notification.

Destination: to the user.

Requires: configuration from garage owner.

Pre-Conditions: camera shootings and transfer live streaming to servers.

Post-Conditions: live stream available to user and notification warning.

8)Logout:

Description and Priority: this function is performed by either car owner or the garage owner.

Inputs: press logout.

Source: either from the car owner or the garage owner.

Outputs: none.

Destination: Out of the system.

Requires: user logged in system.

Pre-Conditions: user logs out.

Post-Conditions: saves everything & logout.

➤ FUNCTIONAL & NONFUNCTIONAL REQUIREMENTS

1. Functional Requirements: -

Car owner requirements: -

- System generates garage available info in two shapes: first one is to search for the nearest garages around his existence place, second one is choosing destination of his ride to also search for available garages.
- When choosing the garage, system display all info about that garage (Photos, Area, Height) and his owner appears.
- When choosing his suitable one, App leads the owner to the garage destination by drawing the route on map.
- System approves two methods of payment Cash or via smart cards.
- A needed screen appears for rating the garage and its services.
- Web camera can be used to check his car.

Garage owner requirements: -

- When car owner choosing his garage, system notify the garage owner of his garage rent operation.
- There are two methods for payment Cash or via smart cards the system should determine the payment method.
- Web camera can be used to check his garage.

A needed screen appears for rating the car owner and his treat.

System Requirements:

- System searches for available garages and display and their info for car owners.
- System drawing the best route for user to reach his garage.
- Ability of renting the garage from a while before getting to it.

2. Non-Functional Requirements: -

- System is accessible 24 hours a day.
- Safety and Security is required so we provide it.
- Response time is acceptable to our customers faith.
- Set a 30-minute timeout to make an account on the system
- Set a timeout to enter your payment data for one hour and then respond to your invoice.
- Providing users with an account and default password.

Smart Parking System

- performance
 - reliability
 - security
 - usability
 - must run on certain platform or operating system
 - must be written in a certain programming language.
- ” Non-functional requirements may be more critical than functional requirements”.

System Requirements:

- The system shall automatically generate the report for printing after every month.

- System cancels the renting after fifteen minutes.
- using camera for reading car number and check the permission of this car.

Organizational requirements:

- System will be implemented by Android, PHP, Web design languages.
- any operating system to work.
- need servers with high storage capacity.
- needs data analysis and Artificial Intelligence.

External requirements:

- system shall implement citizen privacy provisions.
- Cooperation should give its users a license so that no problems with government.

➤ Other Nonfunctional Requirements

■ Performance Requirements:

- This software servers have 1000MB of SSD disk in version 1.0.
- The system guarantee Bandwidth of 10000MB (10GB) for many user accesses at one time.
- Application response time is 3.6 ms and for slow connection it would be 9.8 ms.

■ Security Requirements:

- Passwords must be a minimum of eight characters and must contain one to seven digits.
- E-Mails are unique & must be verified to identify the user this verification made by system that has users in database.

Smart Parking System

- Car owners can't reserve a garage without completing his profile data.
- Users can only chat with others that have deal with.

■ Safety Requirements:

- If car owner's internet connection is lost so he can ask garage owner to have his car out of the garage.
- If there is any movement in the garage, the camera would detect it immediately and push a notification to car owner.
- The lock of the garage would never be opened unless the scanned code is equal to the reservation that the car owner made.
- All data about the users that have reserve and places they reserved is saved in company servers so they can return and check history for any problem.

➤ Software Quality Attributes

- **Usability:** Rakna Guide application is so easy to use and anyone can reserve a garage, we apply best UI/UX for this.
- **Robustness:** The system design shall include recovery scenarios allowing the ability to restore a state no older than one year old.
- **Database:** Is a structured collection of records or data that is stored in a computer system. In our system, this may pertain to garage records or user information.

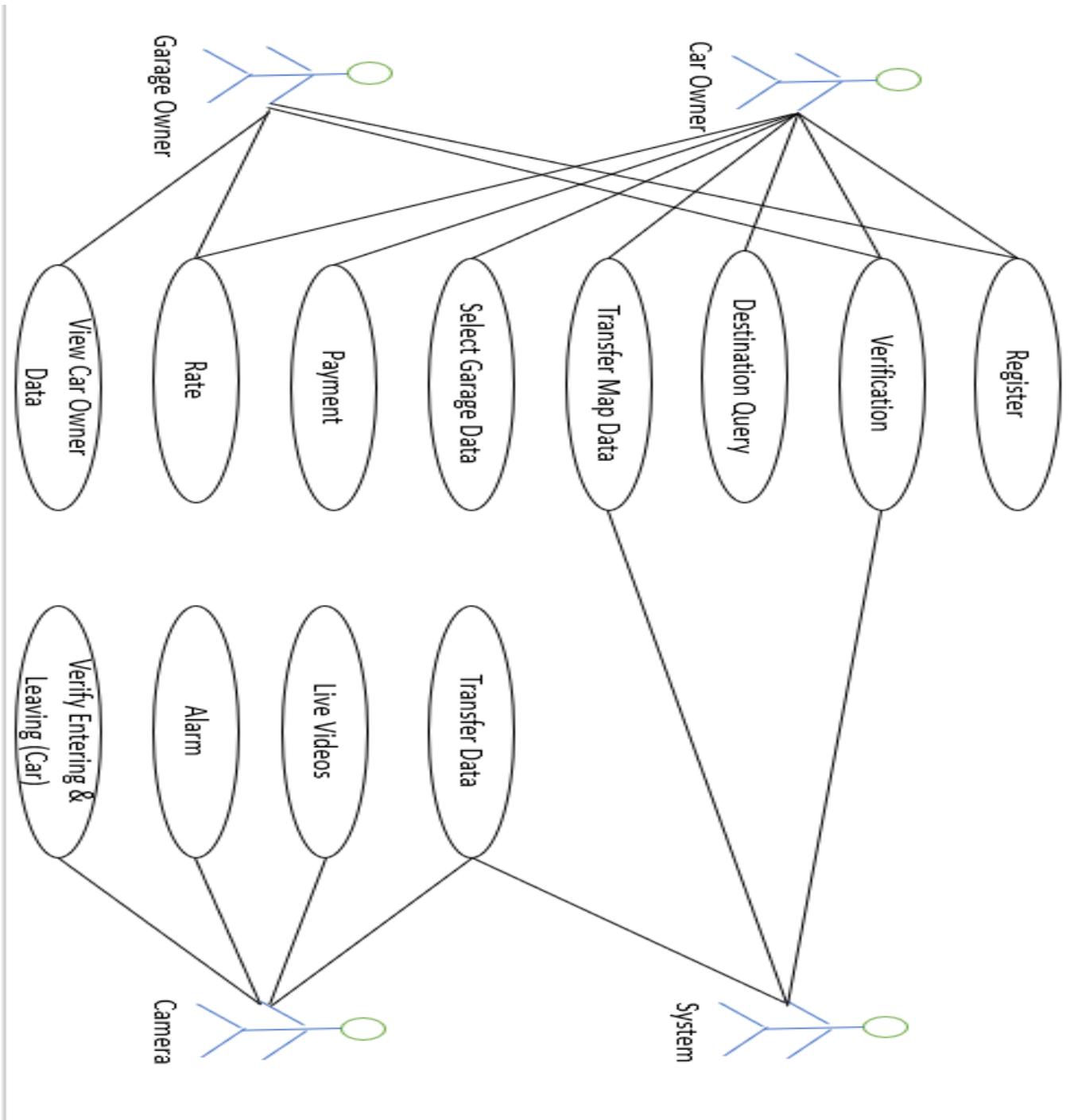
➤ Why this software is important?

Reasons:

- System have a lot of problems such the car owner can't know what garages have an empty space to use.
- It wastes time and effort of car owner to get a nearby garage with empty space.
- There are no safety guarantees if you parked the car under the bridge or with Al-Says and Al-Says would take the car keys to make car maneuvers.
- Car owner has no ability to follow his car situation when he is away.
- Lack of central management can be referred to when a problem occurs.
- There is no data about car owner nor garage owner that we can get back to.

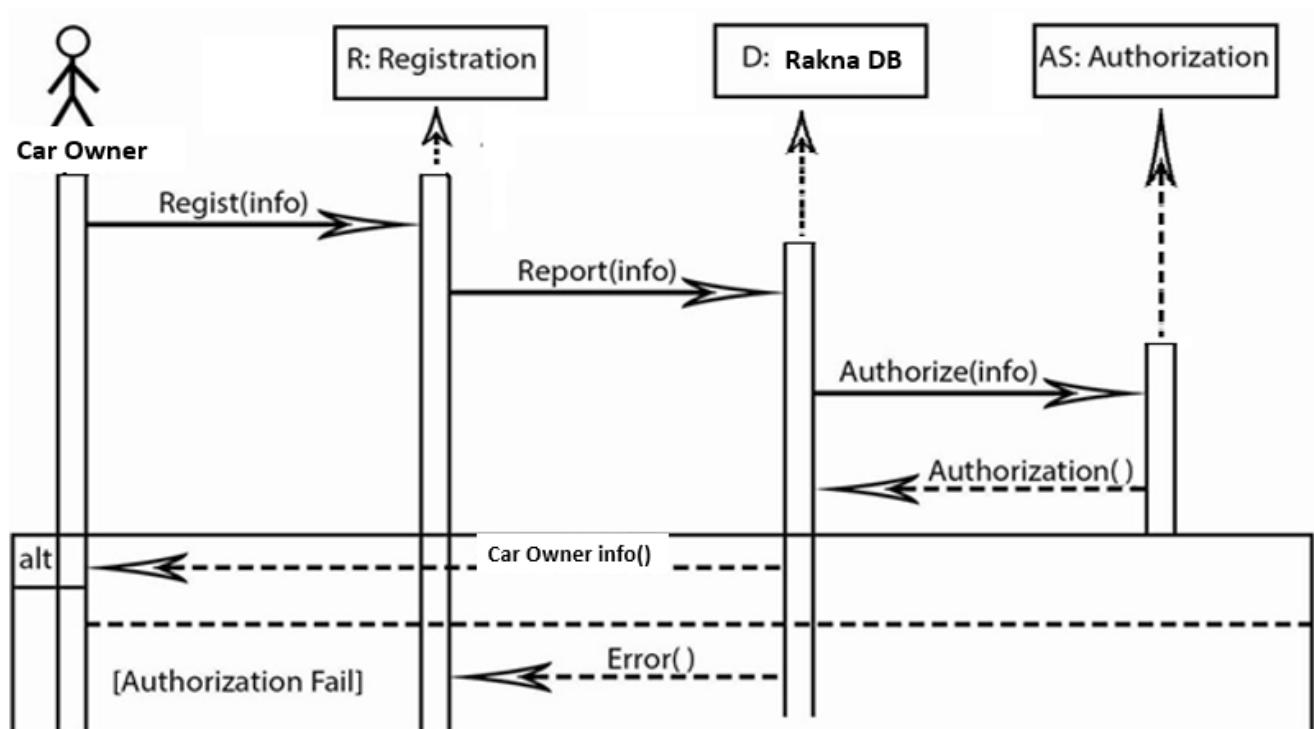
➤ System Modeling

1. Use Case Diagram

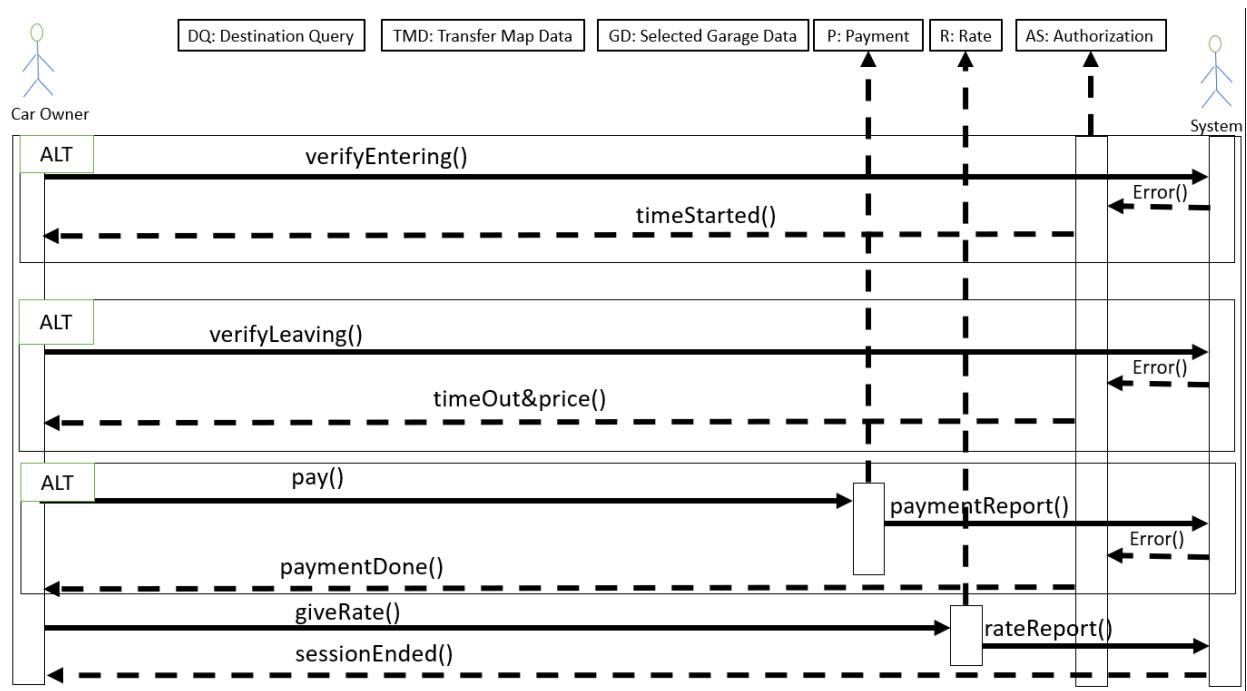
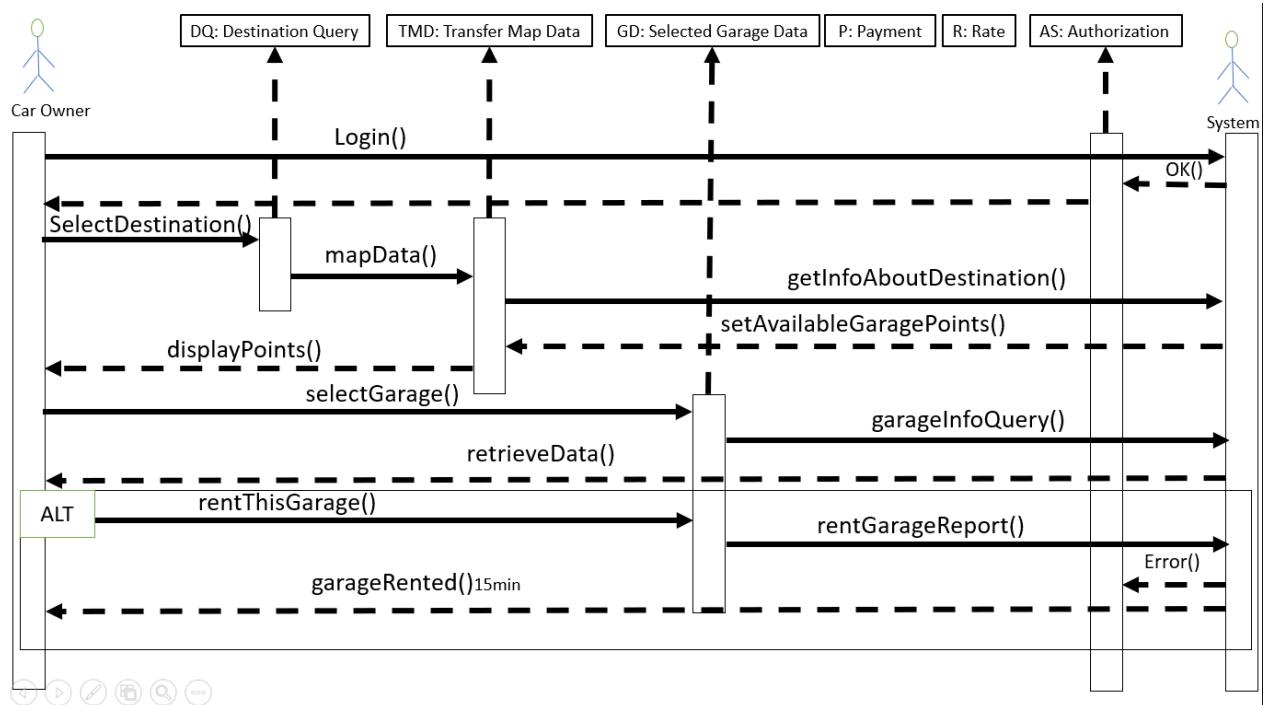


2. Sequence Diagram

2.1 Car owner scenario

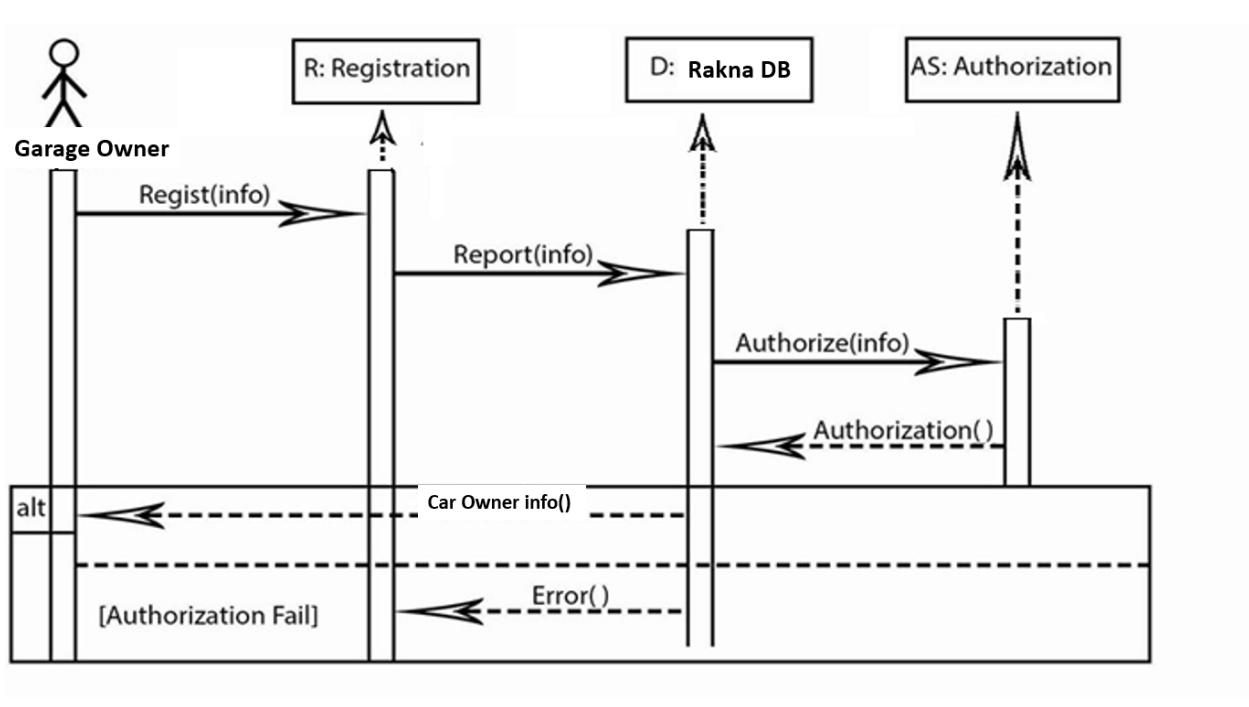
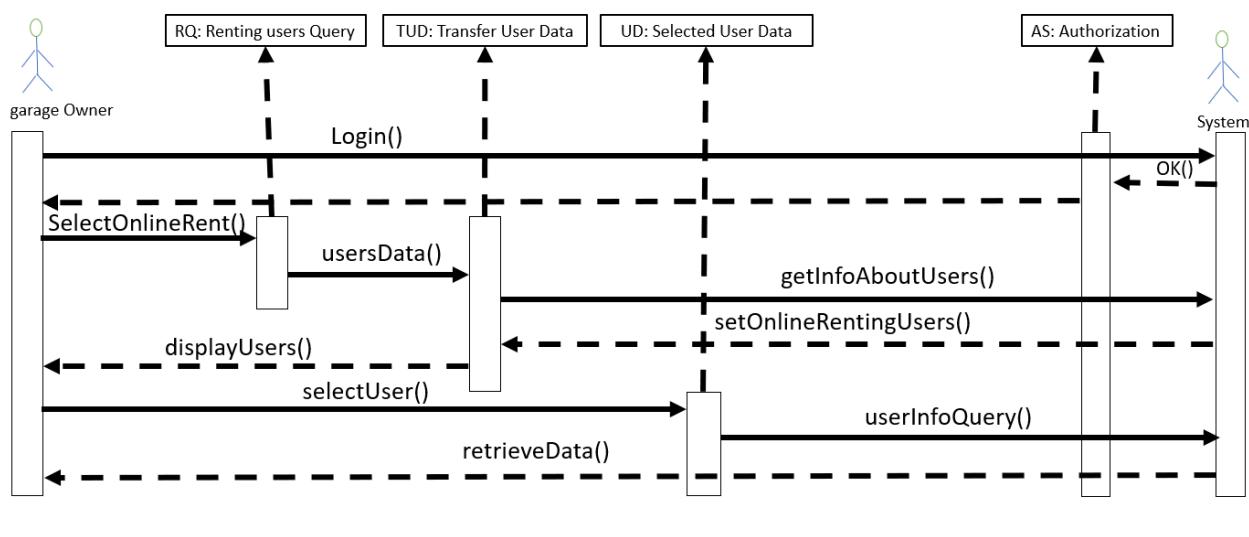


Smart Parking System



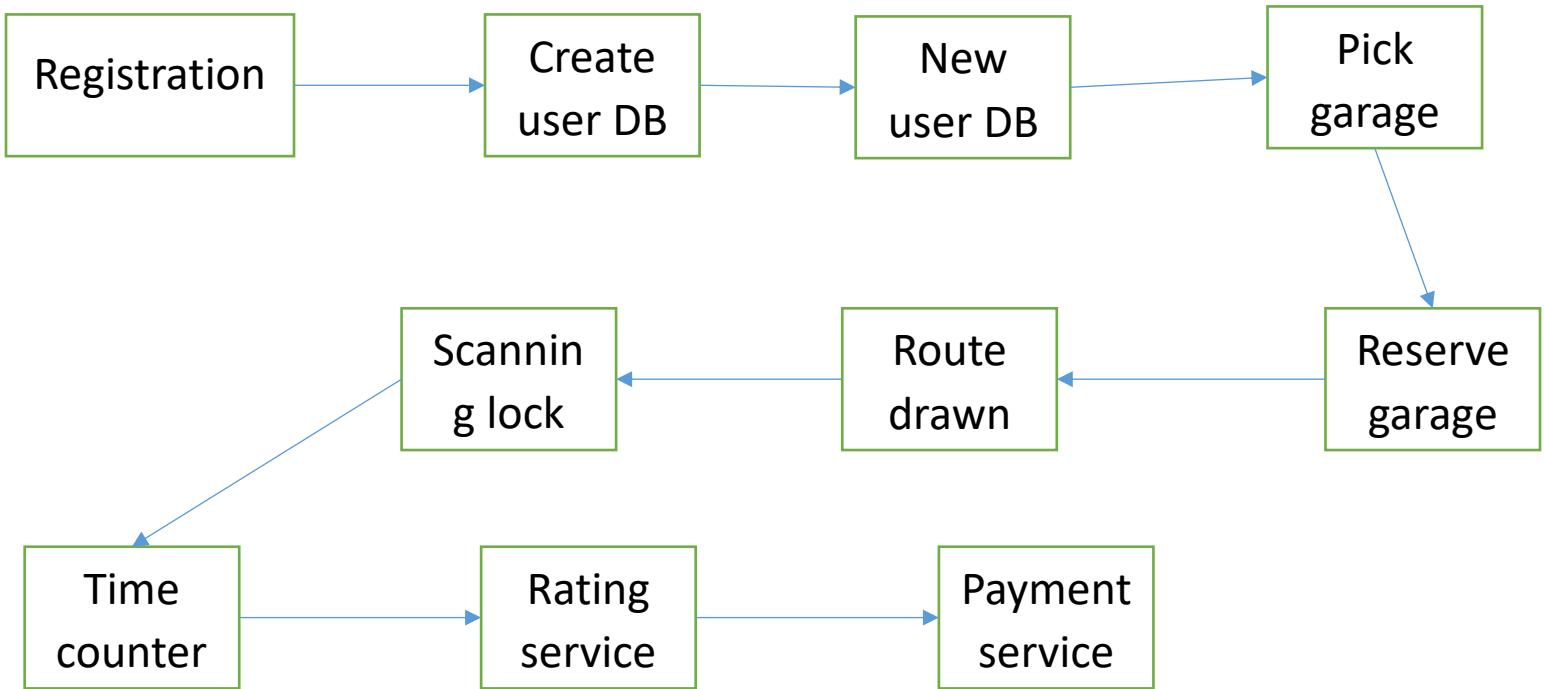
Smart Parking System

2.2- Garage owner scenario

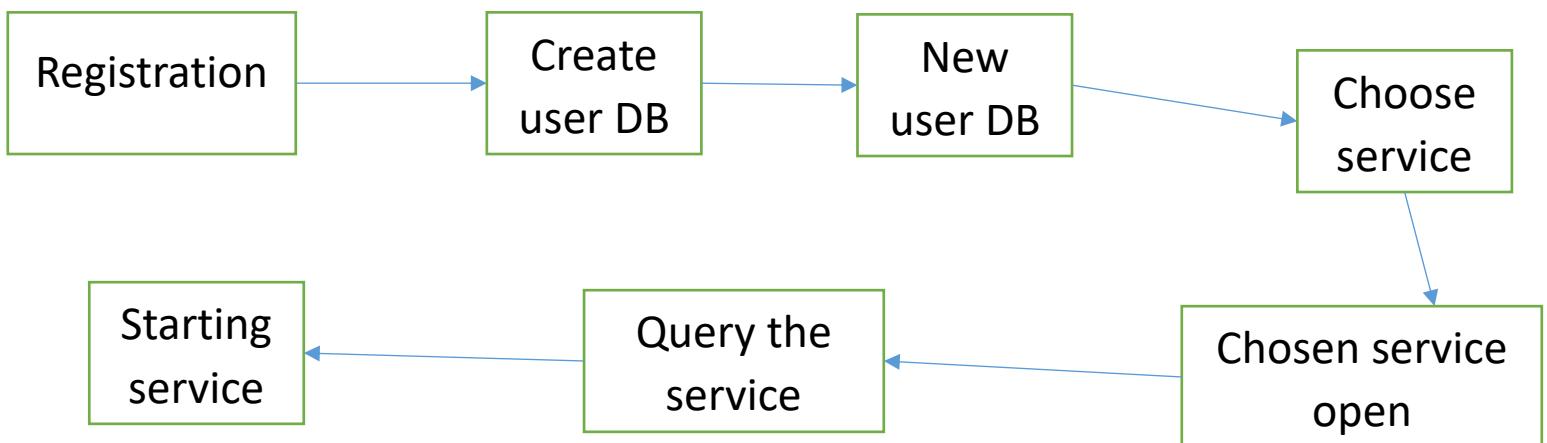


3. Activity Diagram

3.1 Car owner



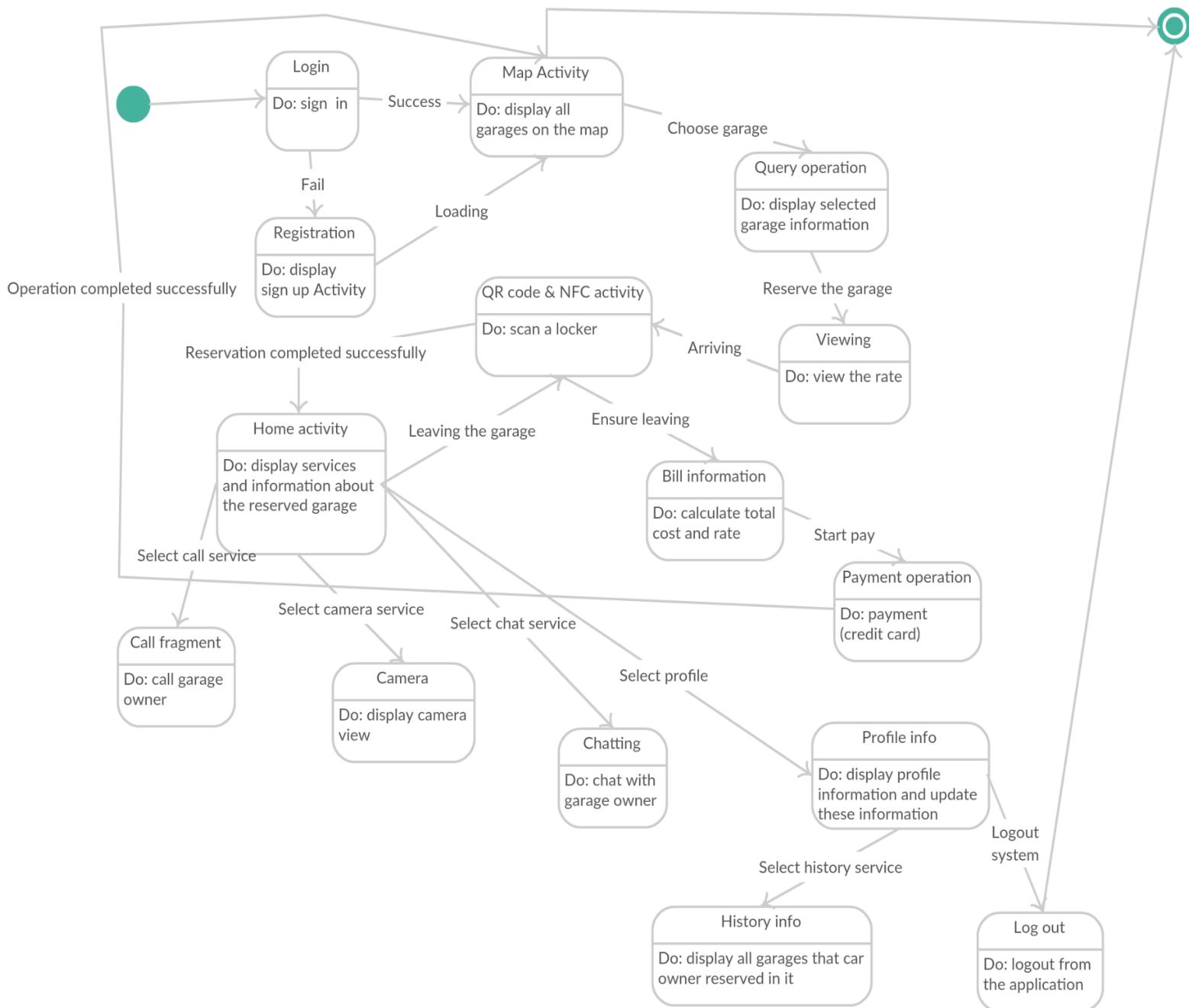
3.2 Garage owner



Smart Parking System

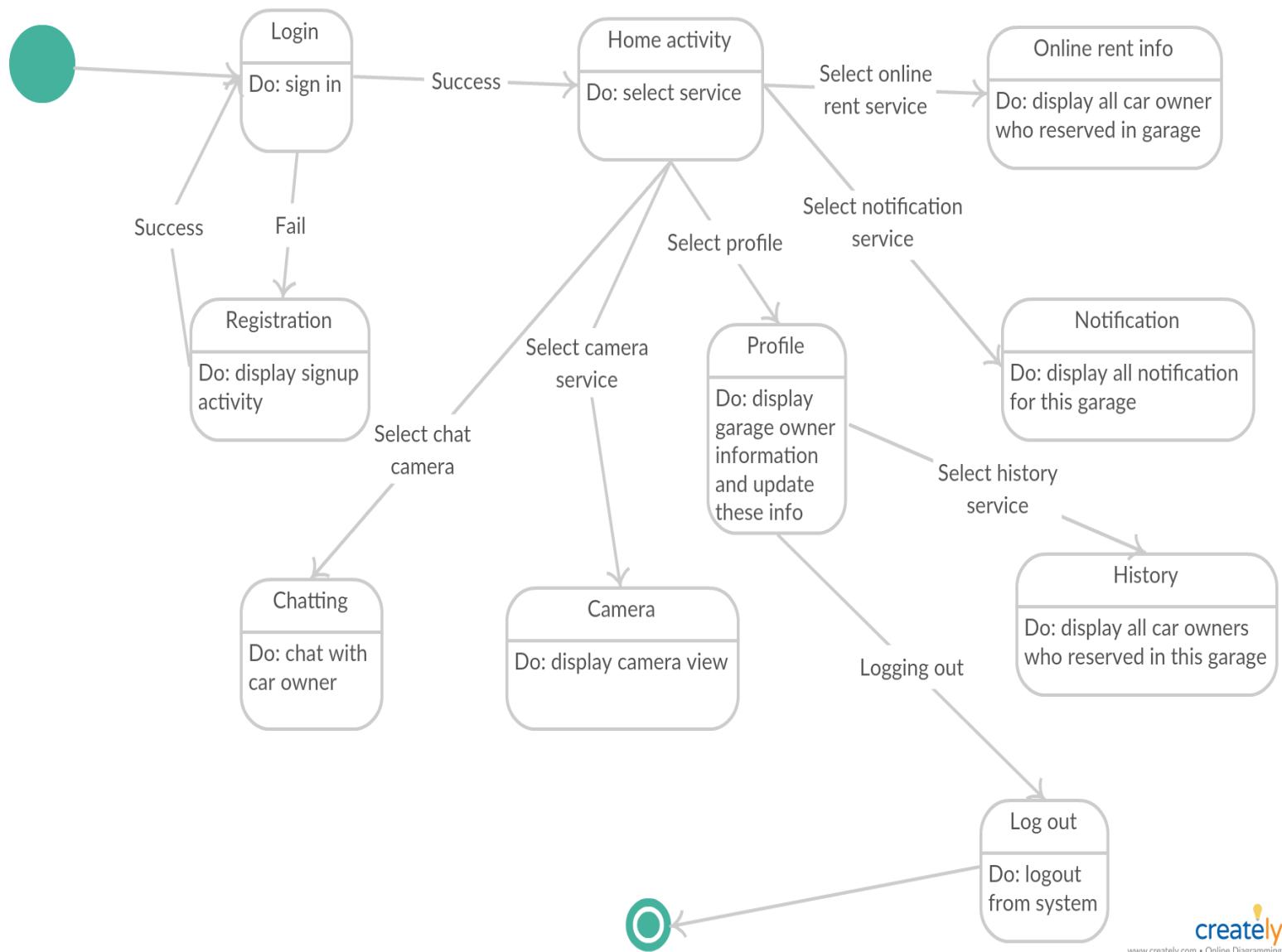
4. State Diagram

4.1 Car owner



Smart Parking System

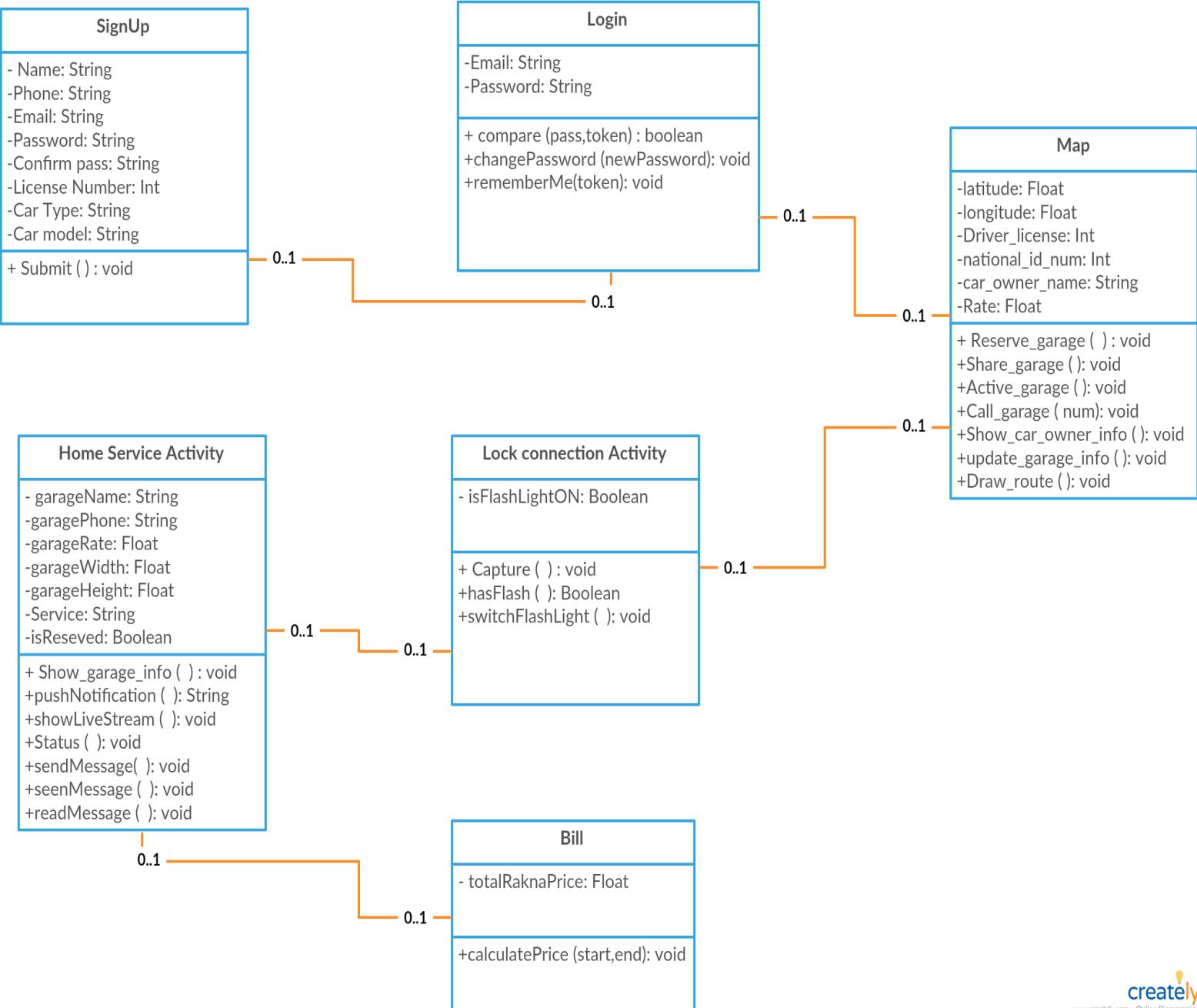
4.2 Garage owner



Smart Parking System

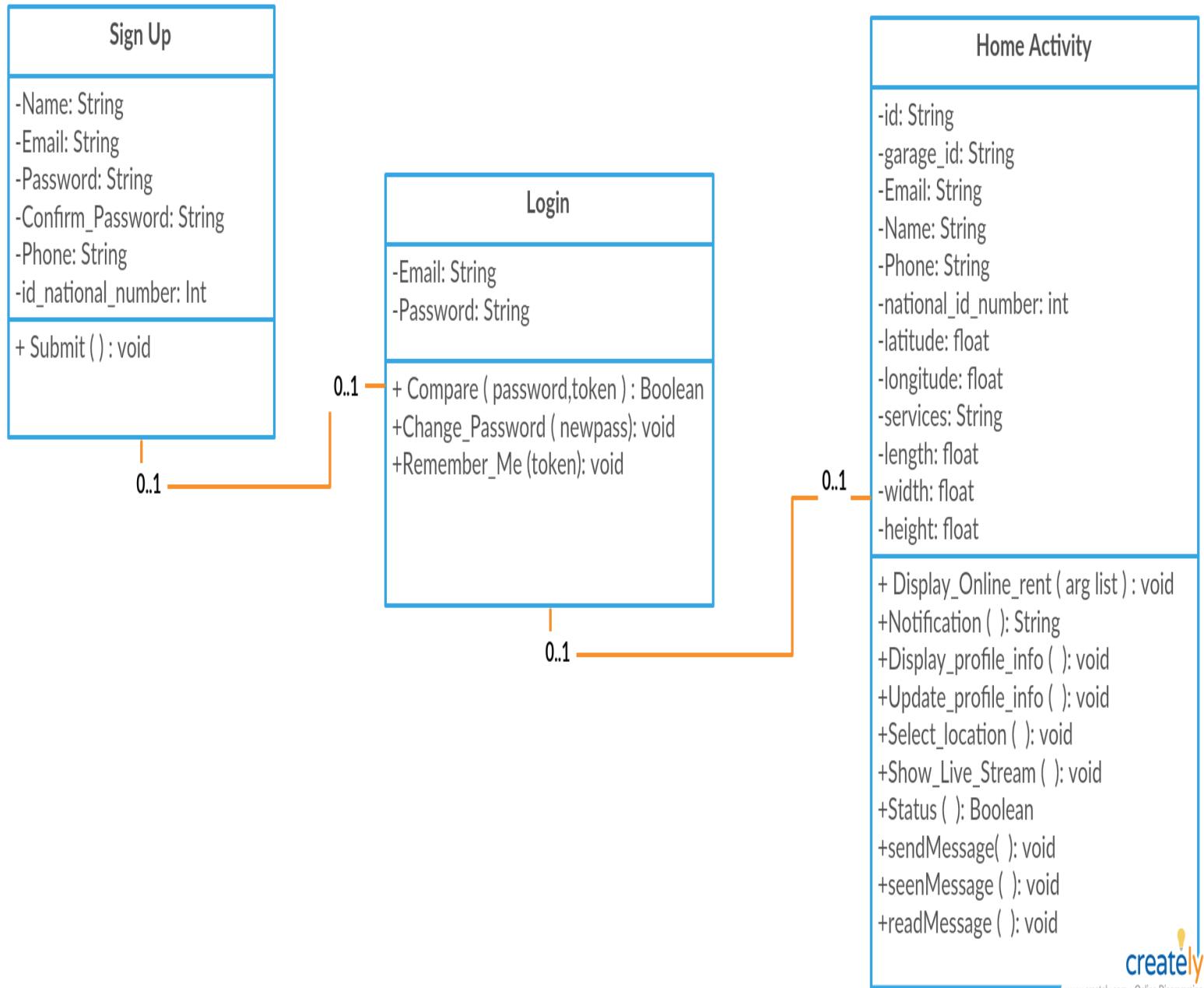
5. Class Diagram

5.1 Car owner



Smart Parking System

5.2 Garage owner



➤ Client - Server Architecture

Name	Client-server
Description	<p>In a client server architecture, the functionality of the System is organized into our services that related to payment methods. With each service delivered from a separate Server. Clients are users of these services & access servers to make use of them.</p>
When use	<p>Used when data in a shared database has to be accessed from a range of locations because servers can be replicated, may also be used when the load on a system is a variable</p>
Advantages	<ul style="list-style-type: none"> - The principal advantages of this model are that servers can be distributed across a network - General functioned is payment services can be available to all clients & does not need to be implemented by all services
Disadvantages	<ul style="list-style-type: none"> - Each service is a single point of failure so susceptible to denial of service attacks (DOS) or server failure. - Performance may be un predictable because it depends on the network as well as the system may be management problems if servers are owned by different organizations.

➤ Test Cases

1- CAR OWNER TEST CASES:

Test Case ID	Test Scenario	Expected Result	Test Parameters
Activity 1:name	Verify that the input field accept maximum number of 30 characters full name with no special characters	Application should be able to accept all 30 characters with no special characters	<ul style="list-style-type: none"> - Full name with no special characters - Name with numbers and characters - Number of characters are greater than 30
Activity 1:phone	Verify that the input field accept maximum number of country key accepted phone number (eg: egypy 002+11 numbers of phone)	Should be able to accept all numbers depends on country key	<ul style="list-style-type: none"> - Numbers with correct length and correct country key - Numbers don't match with country key or incorrect length
Activity 1:car license (camera)	Verify that the input field accept maximum number of country license (eg:2861 ج ٢١)	Should be able to accept all numbers depends on country vehicles license(numbers and letters)	<ul style="list-style-type: none"> - Numbers and letters with correct length - Numbers without letters or vice versa
Activity 1:driver license	Verify that the input field accept maximum number of 14 numbers (eg:ID)	Should be able to accept all numbers depends on ID system in each country	<ul style="list-style-type: none"> - Numbers with correct length - Numbers and letters

Smart Parking System

activity 1:email	verify that the input field accept strings with special characters @ (eg: gmail, yahoo) the must be dot com (eg: .com)	Should be able to accept all emails with special characters @ (eg: gmail, yahoo) the must be dot com (eg: .com)	<ul style="list-style-type: none"> - Correct email - No dot com - Not available email host
Activity 1:password/confirm password	verify that the input field accept a password with special chars , upper case and lower case , chars and numbers	Should be able to accept all password with special chars , upper case and lower case , chars and numbers	<ul style="list-style-type: none"> - Sequence numbers - Contains person name
activity 1: verification code	verify that the input field accept four digits	Should be able to accept the sent four digits	<ul style="list-style-type: none"> - Characters - Unavailable digits - Correct digits
Activity 6:credit number			-

2- GARAGE TEST CASES

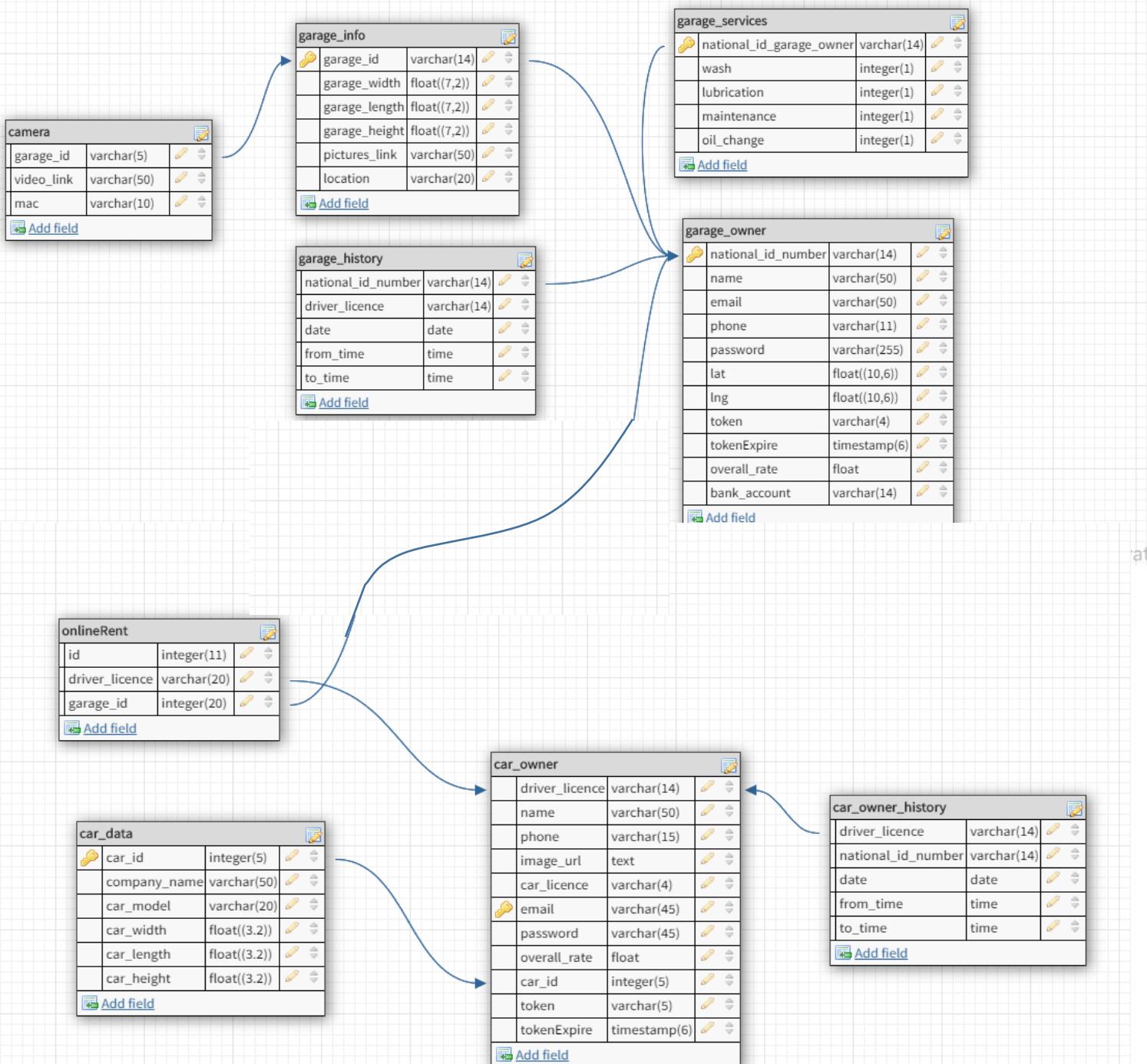
Test Case ID	Test Scenario	Expected Result	Test Parameters
Activity 1:name	Verify that the input field accept maximum number of 30 characters full name with no special characters	Application should be able to accept all 30 characters with no special characters	<ul style="list-style-type: none"> - Full name with no special characters - Name with numbers and characters - Number of characters are greater than 30
Activity 1:phone	Verify that the input field accept	Should be able to accept all numbers	<ul style="list-style-type: none"> - Numbers with correct length and correct country key

Smart Parking System

	maximum number of country key accepted phone number (eg: egypy 002+11 numbers of phone)	depends on country key	<ul style="list-style-type: none"> - Numbers don't match with country key or incorrect length
Activity 1:ID number	Verify that the input field accept maximum number of 14 numbers (eg:ID)	Should be able to accept all numbers depends on ID system in each country	<ul style="list-style-type: none"> - Numbers with correct length - Numbers and letters
activity 1:email	verify that the input field accept strings with special characters @(eg:gmail, yahoo) the must be dot com(eg: .com)	Should be able to accept all emails with special characters @ (eg:gmail, yahoo) the must be dot com(eg: .com)	<ul style="list-style-type: none"> - Correct email - No dot com - Not available email host
Activity 1:password/confirm password	verify that the input field accept a password with special chars ,upper case and lower case , chars and numbers	Should be able to accept all password with special chars ,upper case and lower case , chars and numbers	<ul style="list-style-type: none"> - Sequence numbers - Contains person name
activity 1: verification code	verify that the input field accept four digits	Should be able to accept the sent four digits	<ul style="list-style-type: none"> - Characters - Unavailable digits - Correct digits
Activity 1:width/height/length	verify that the input field accept 3 digits in meter	Should be able to accept the sent 3 digits in meter	<ul style="list-style-type: none"> - Enter 3 digits - Enter more than 3 digits - Enter characters

Smart Parking System

➤ Database Design



➤ Architectural Pattern

The pattern used in the system are two combined

- Client-Server pattern.
- Model-View-Presenter (MVP) pattern.

As the types of users acting exactly as Client & server

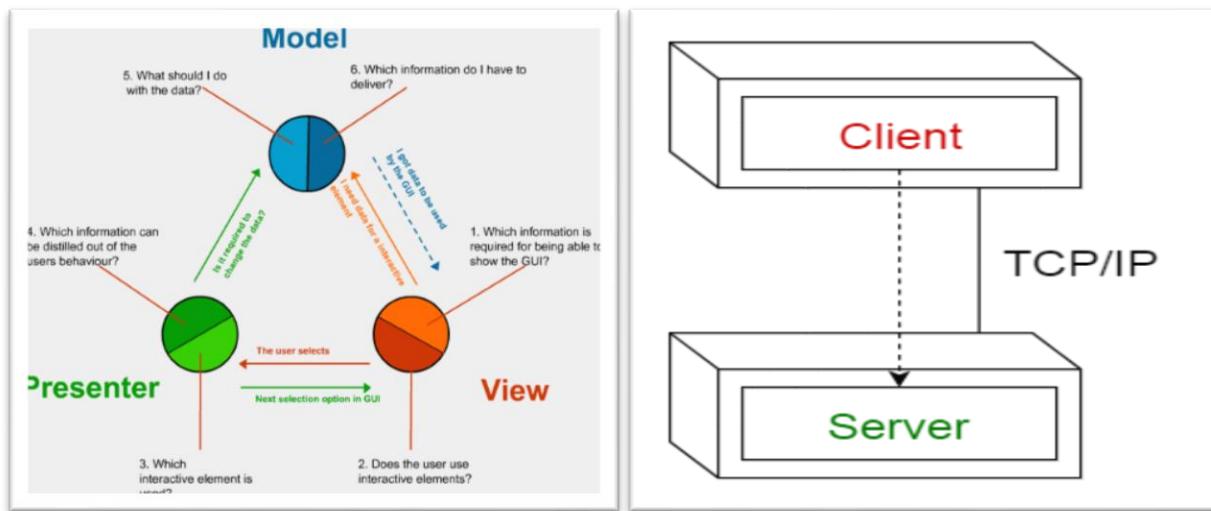
The System is separated into two parts: -

- Car owner application
- Garage owner application

But each of them has its own functionalities & behaviors so we can just say they are 2 projects

1. car owner
2. Garage owner

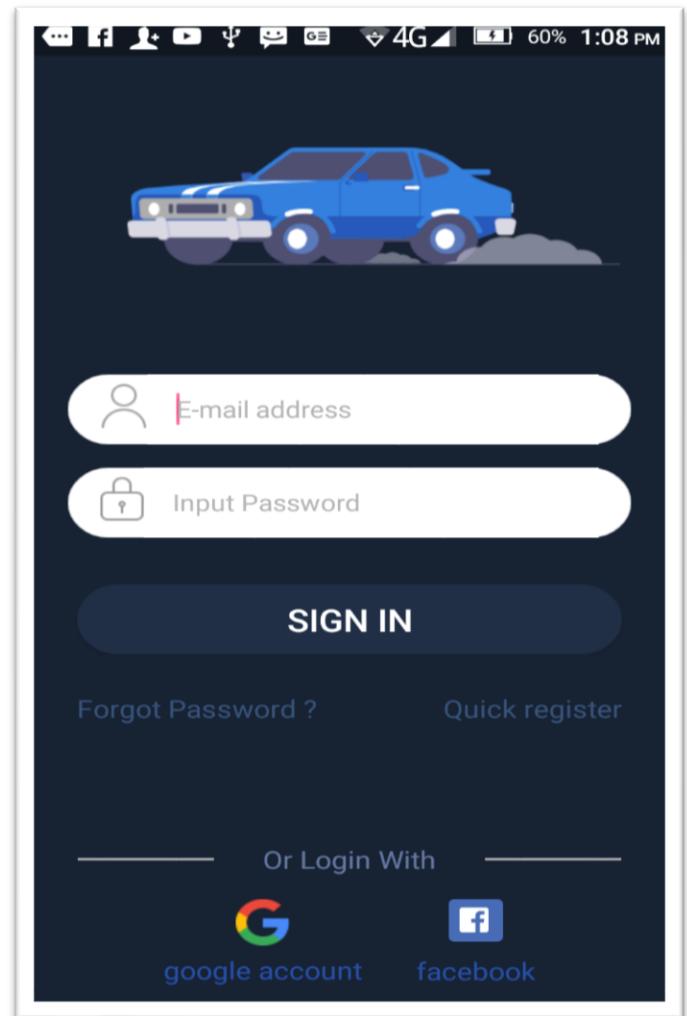
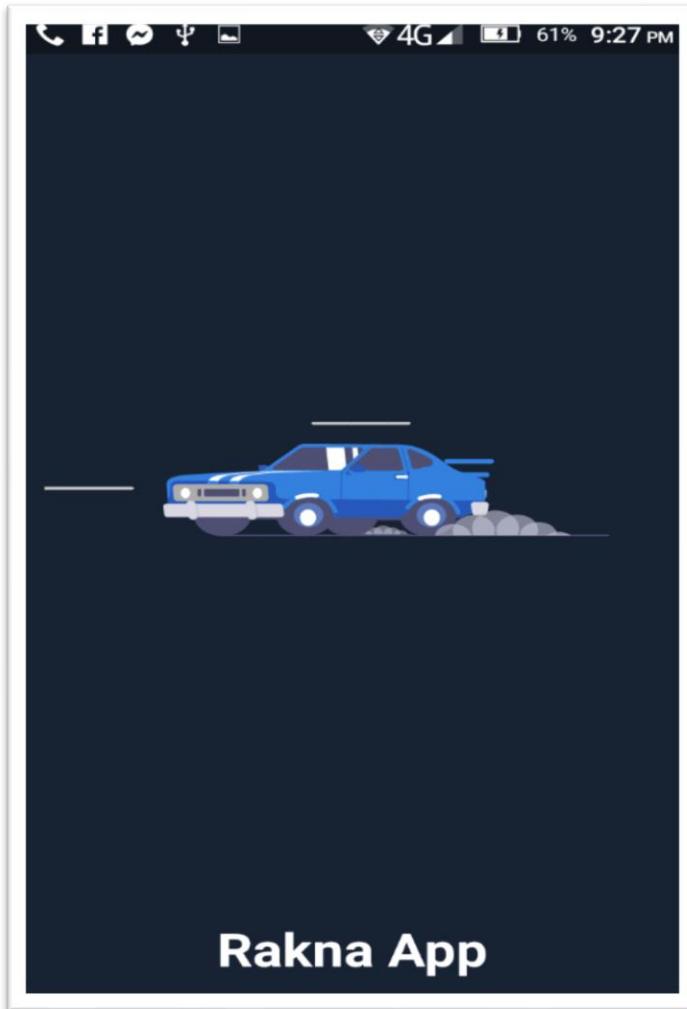
- the pattern used in each of these projects is MVP and Client Server



➤ Development & Implementation

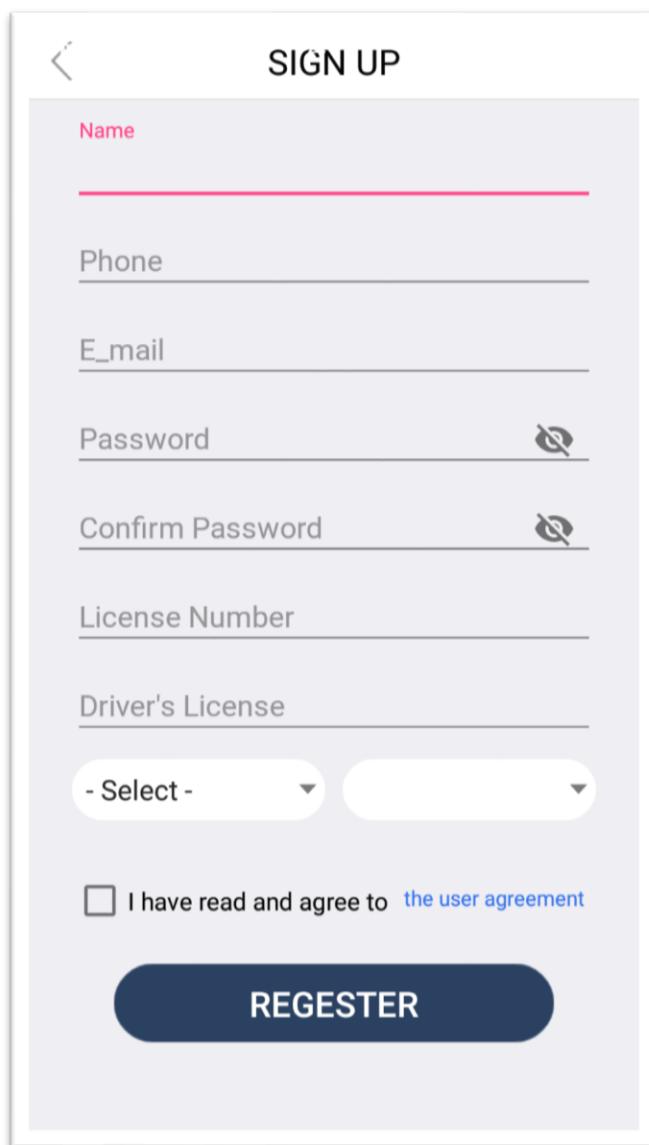
In this phase after doing all the analysis, studies & extracting ideas. We get to the Implementation & coding phase which is considered to be the face of the entire project. So, we'll take a look at each version of the project.

The car owner application



Smart Parking System

The first screen here is the Splash



SIGN UP

Name

Phone

E_mail

Password 

Confirm Password 

License Number

Driver's License

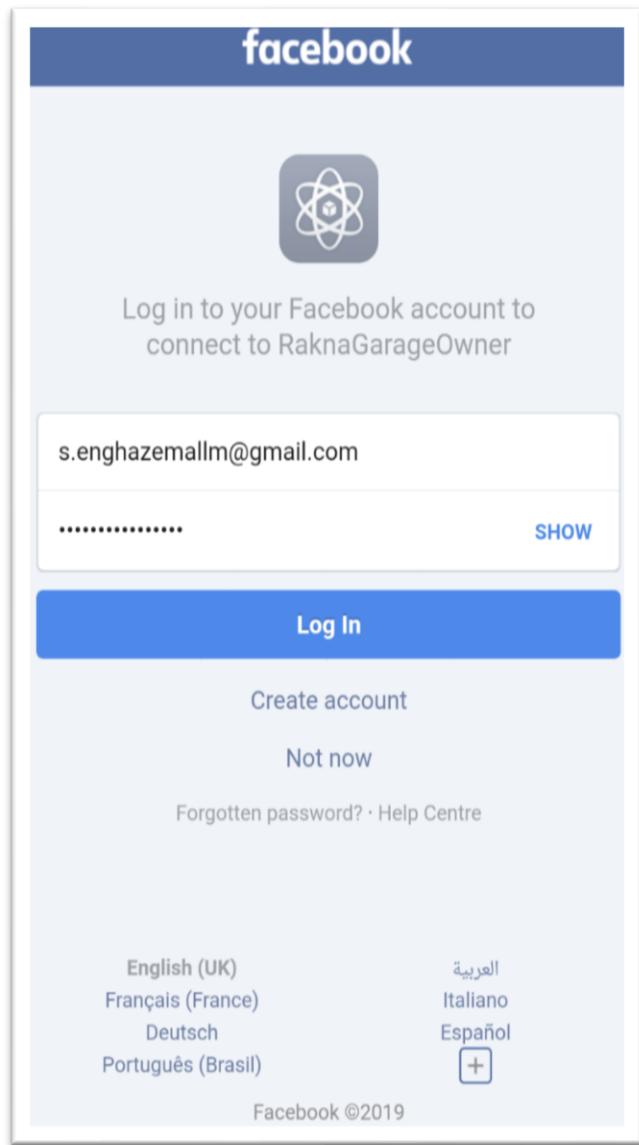
- Select -

I have read and agree to [the user agreement](#)

REGISTER

The next is Login screen

If the car owner is registered before, he can login directly

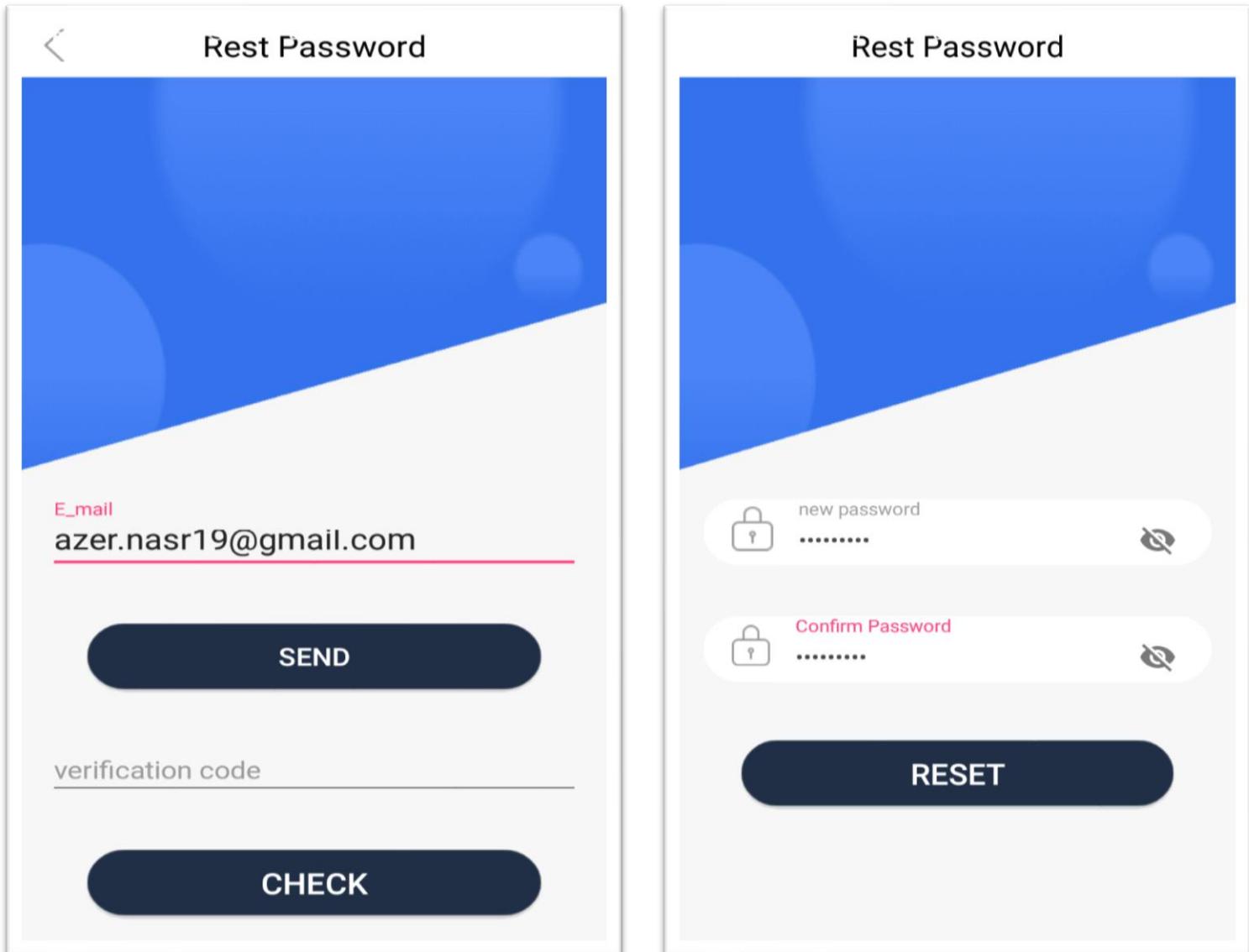


The next is Registration Screen

If the car owner isn't registered before, he can make quick registration by filling his personal details

OR you can login with Facebook or Gmail

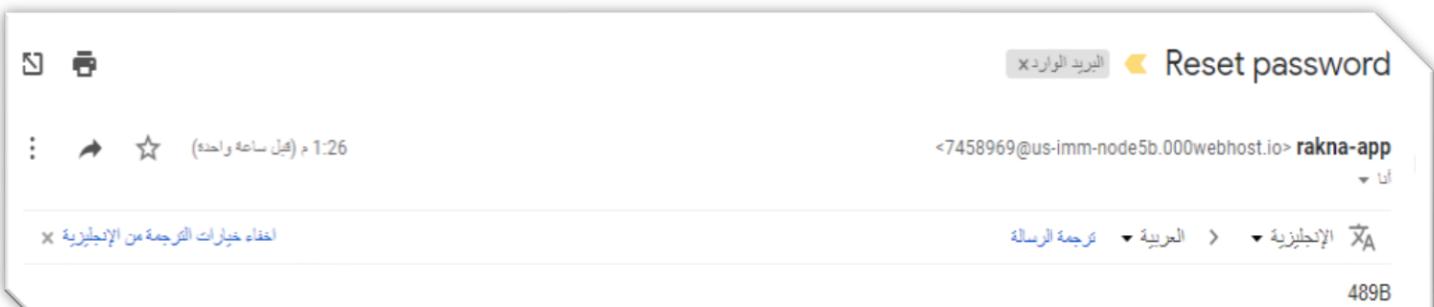
Smart Parking System



Forget password screens

If the car owner forgets his password, he can write his email and receive a token to ensure that this account

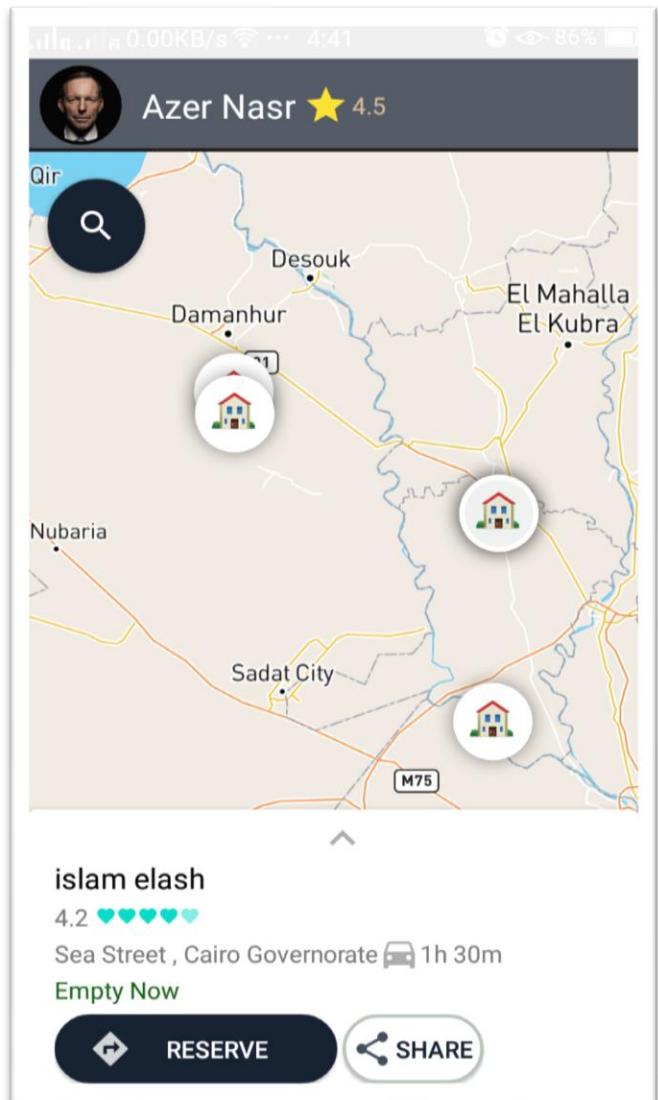
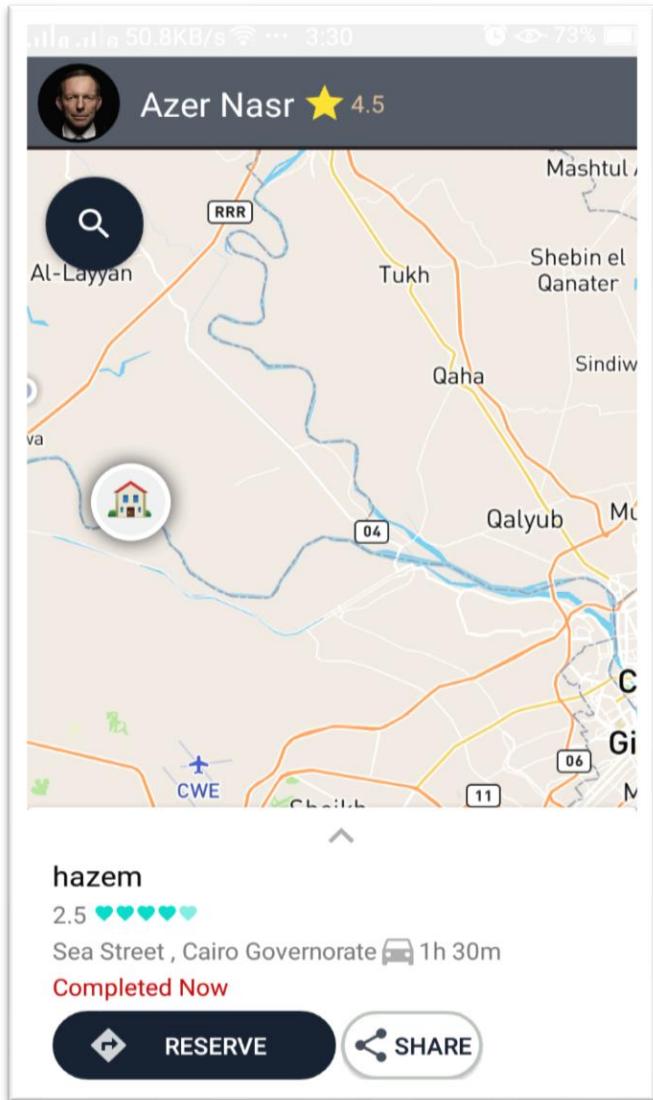
Enter the token, if it right he will enter new password and confirm it



Smart Parking System

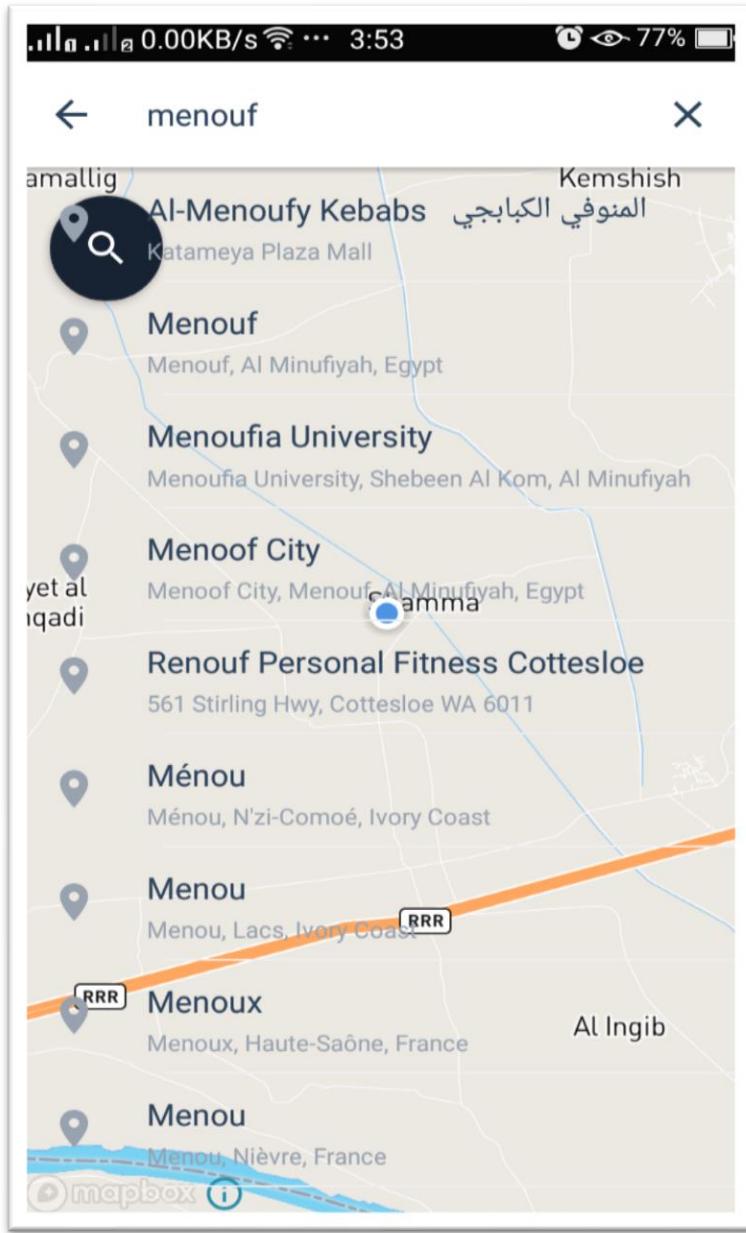
• After Login

The home screen is the map to show all garages locations.

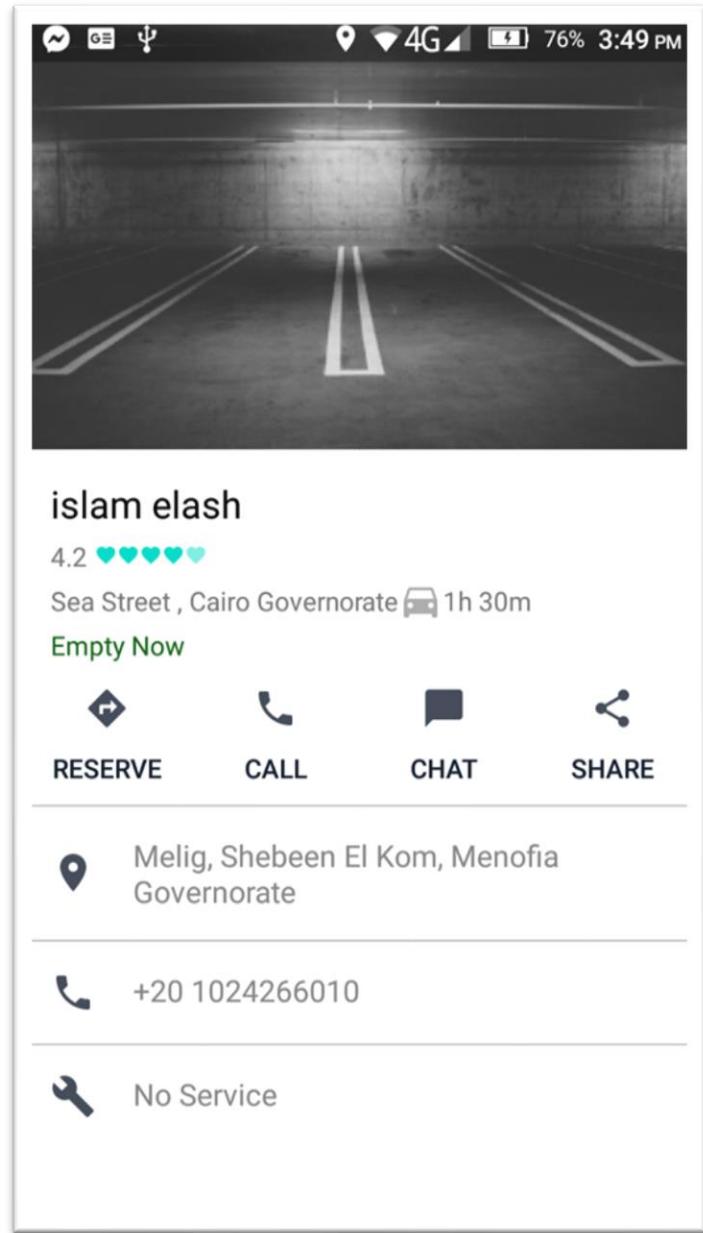


The car owner can select any garage to show details about this garage.

Smart Parking System

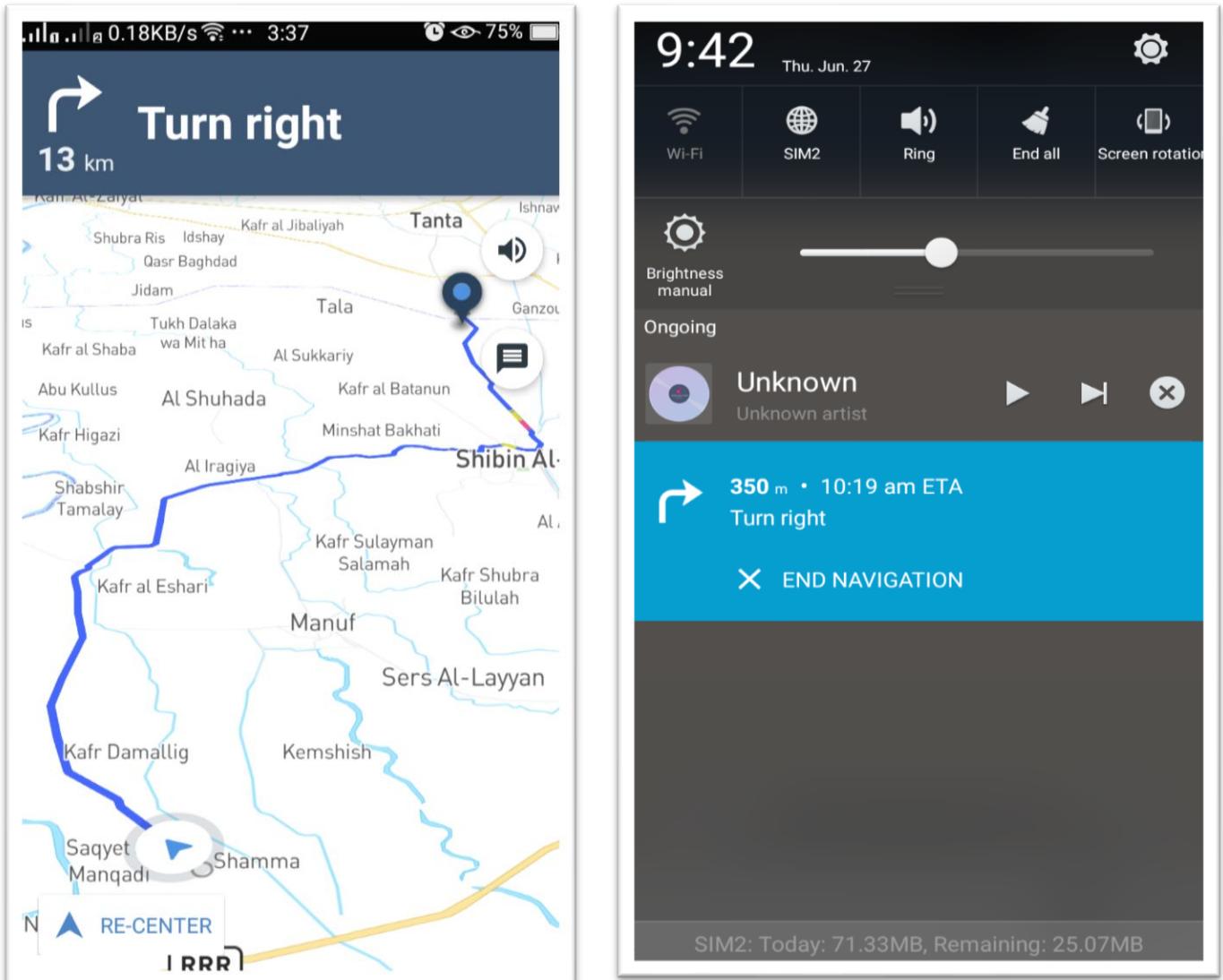


The car owner can use the search engine to find all garages in specific location.



This activity view data about garage owner

Smart Parking System

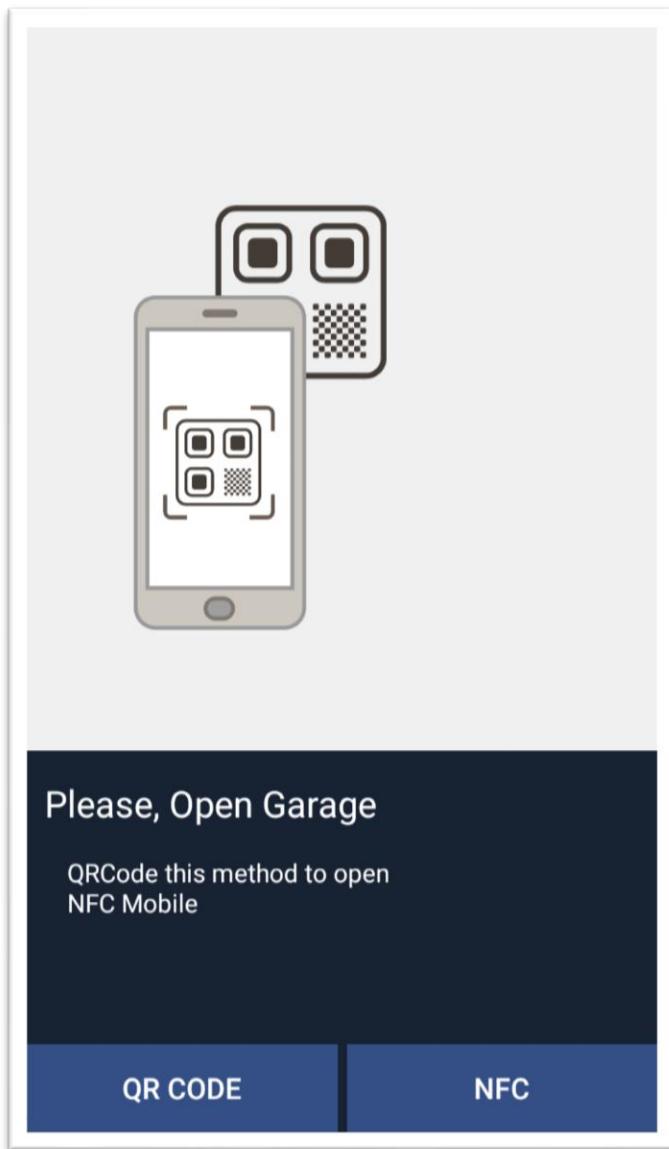


After reserve the garage, the route to it will be drawn.

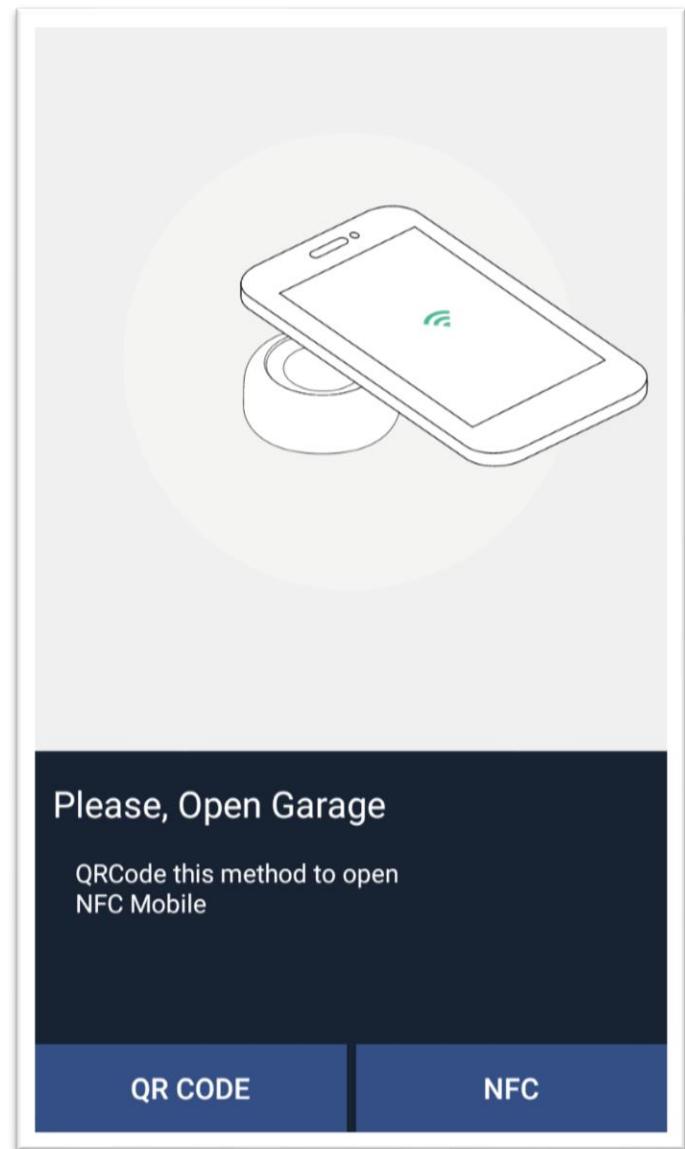
And receive a notification to show the amount of time to arrive this garage.

Smart Parking System

- After car owner arriving, he can click the REACH button to open the garage by NFC or QR code.



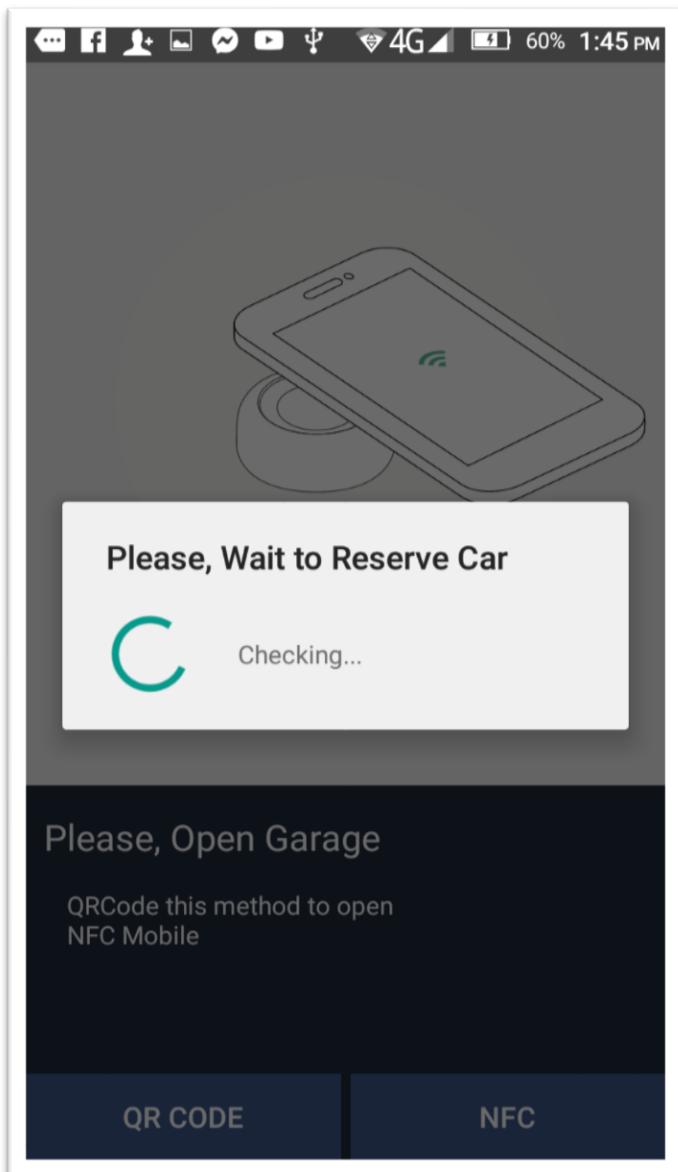
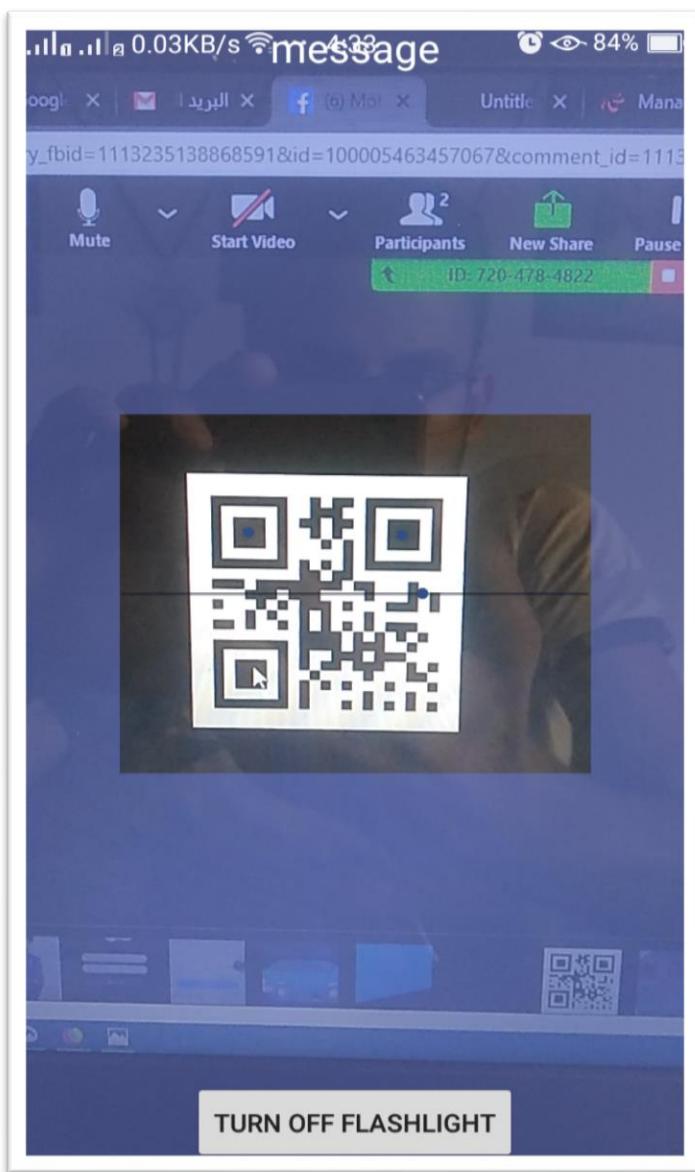
QR code



NFC

- The next QR code or NFC screen

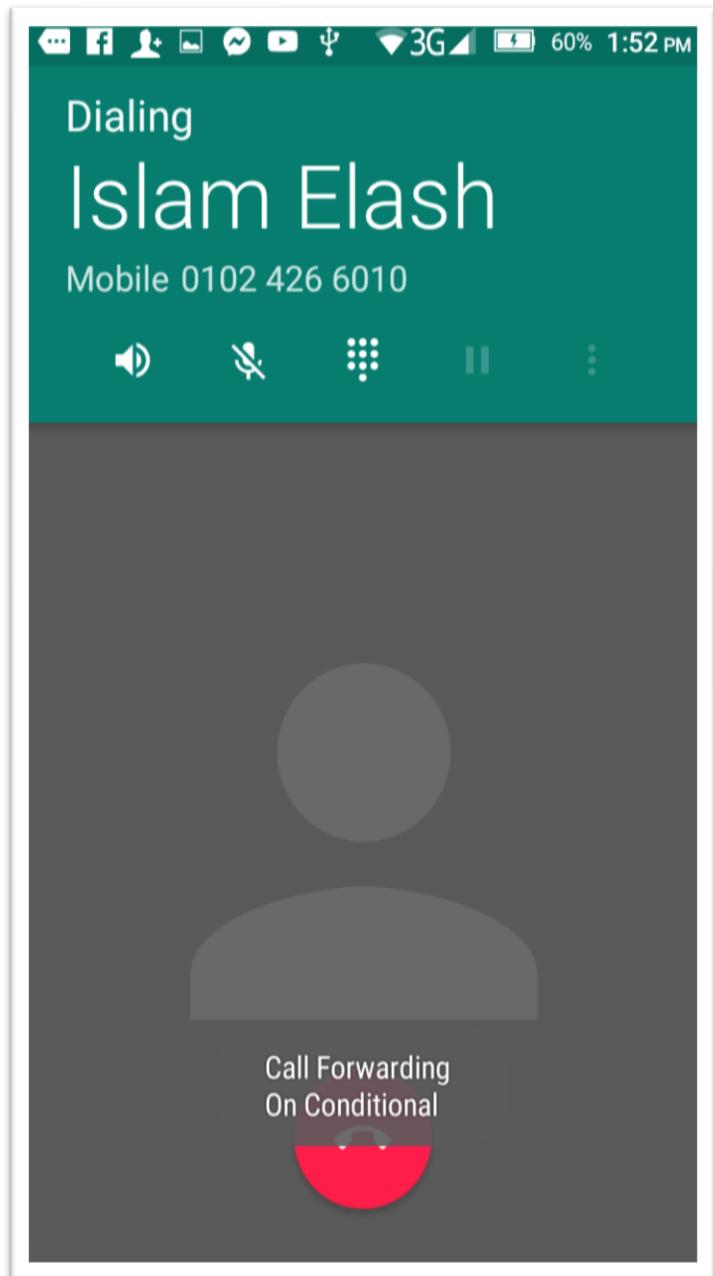
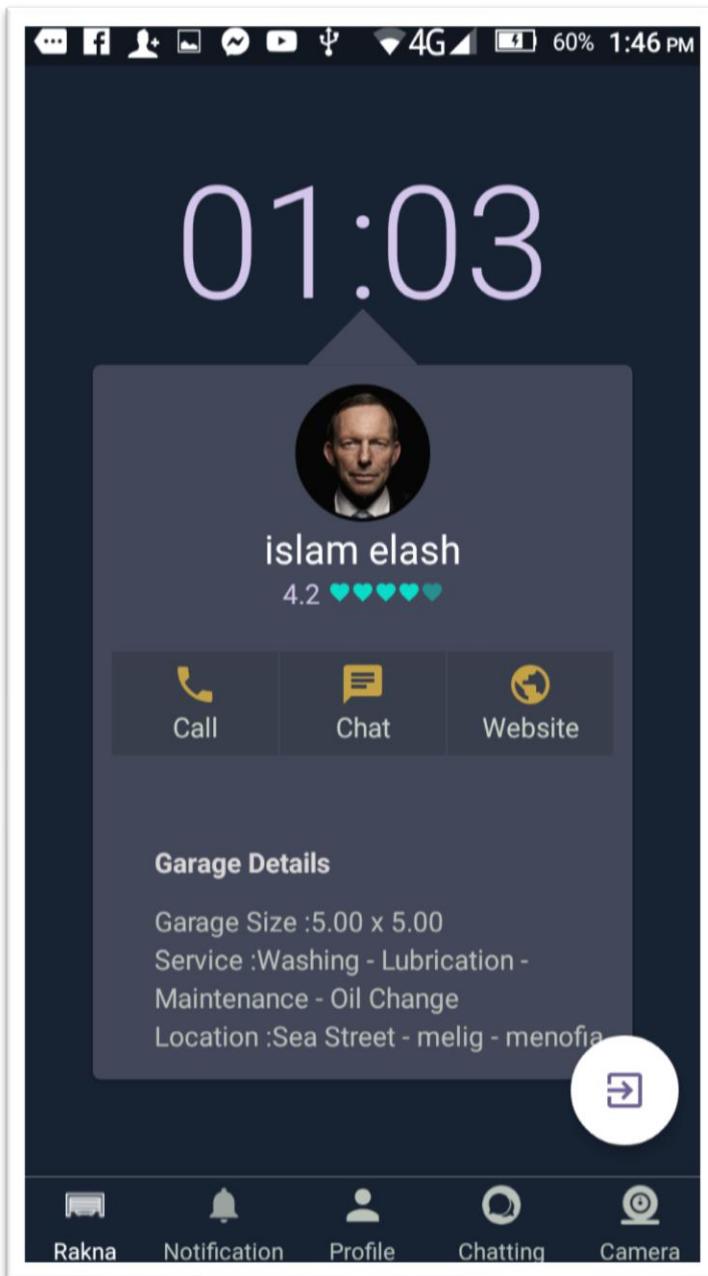
QR code or NFC is used to open the garage but NFC screen can't be launch if the phone doesn't support it.



Smart Parking System

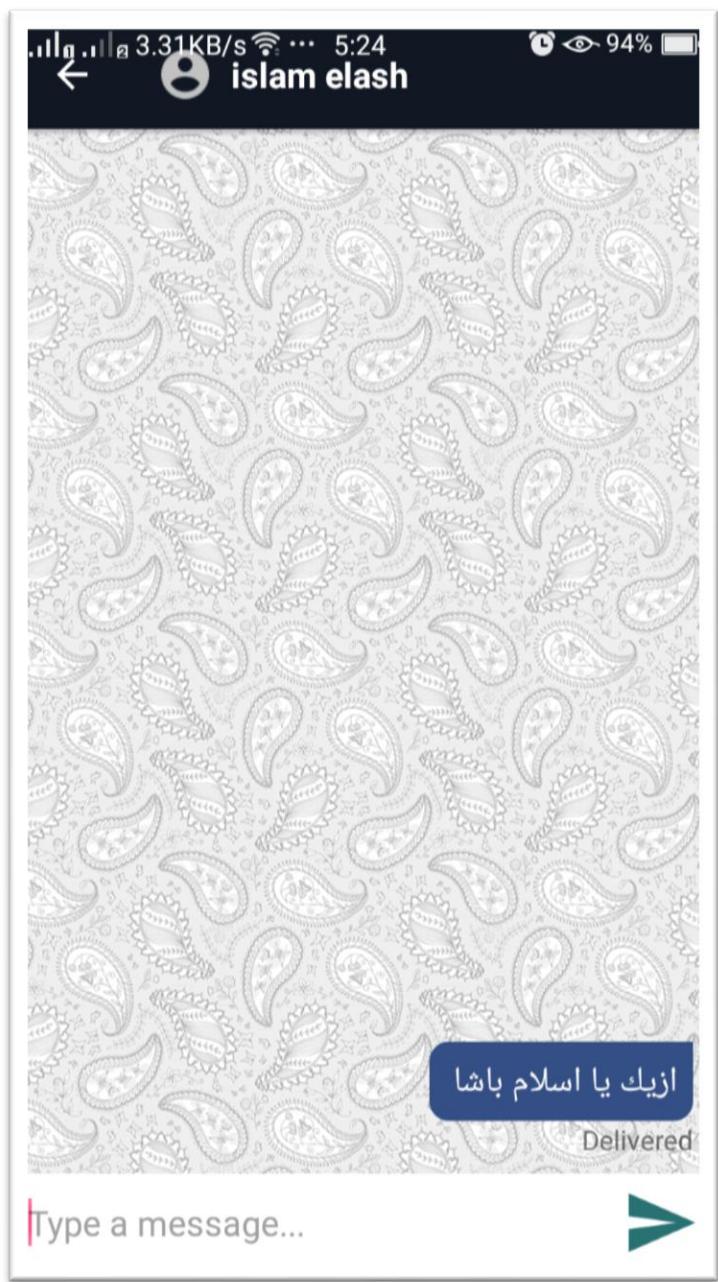
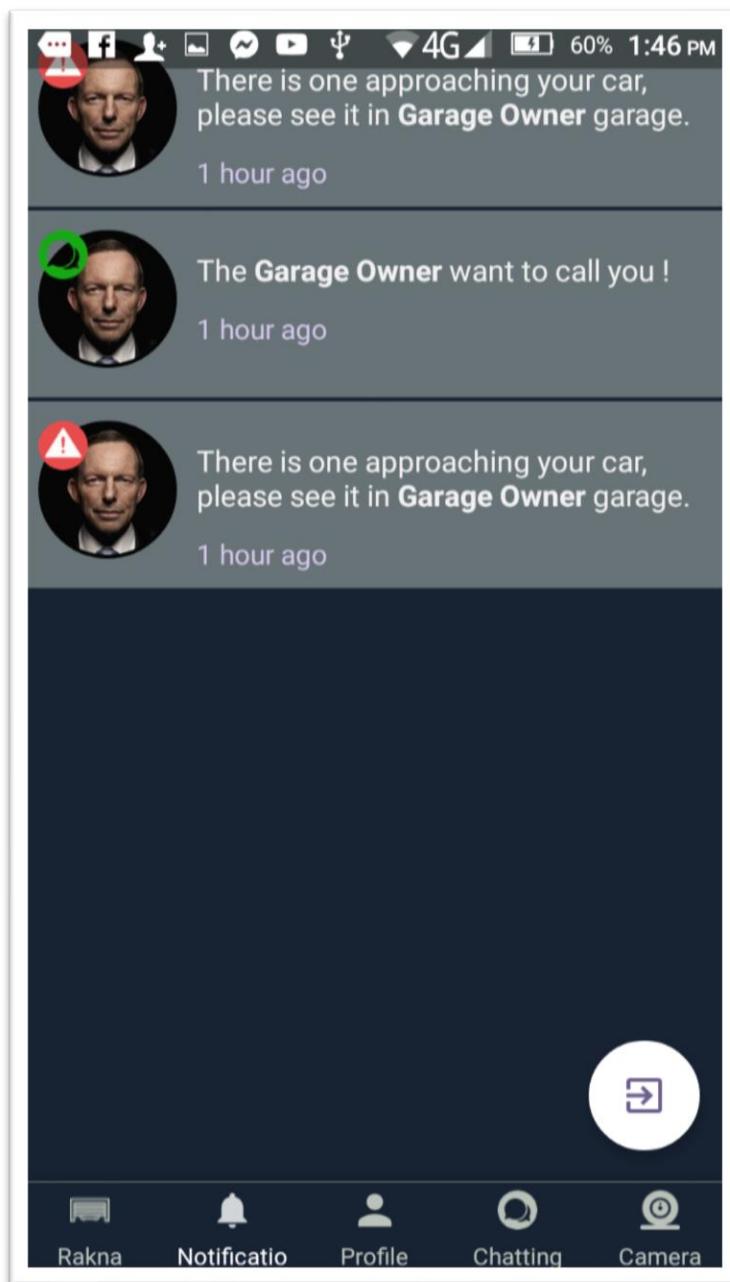
● Next the services screens

First screen contains garage data and services.



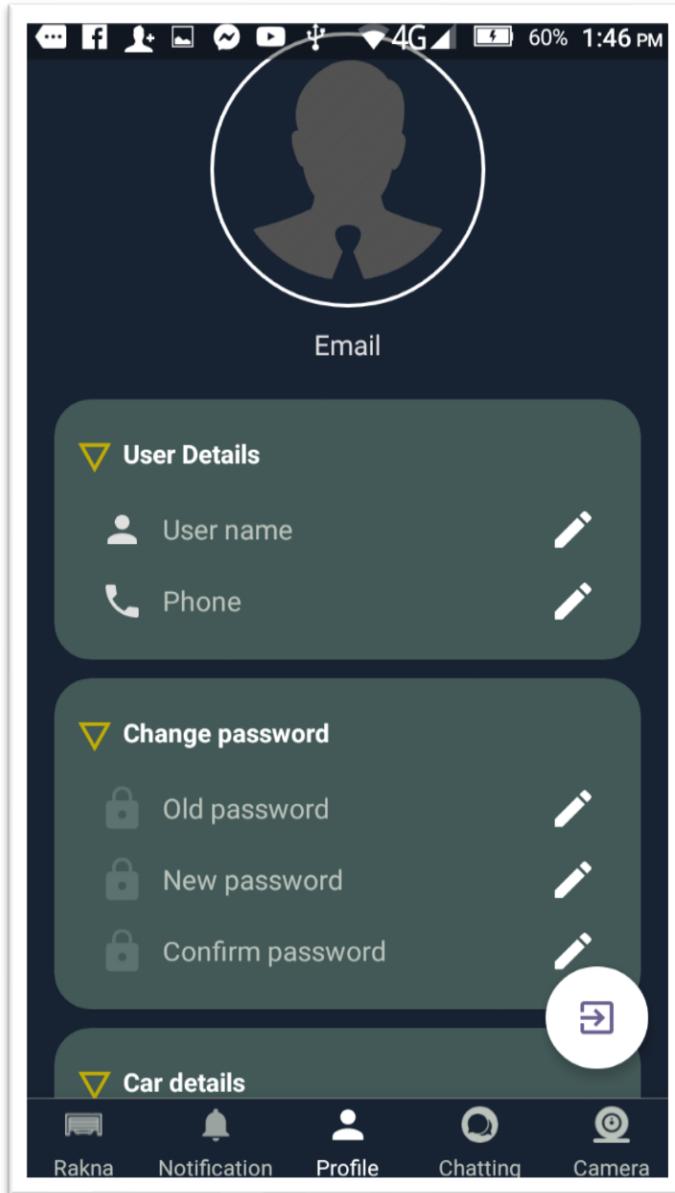
Smart Parking System

- The second screen contains warning notifications and chatting .

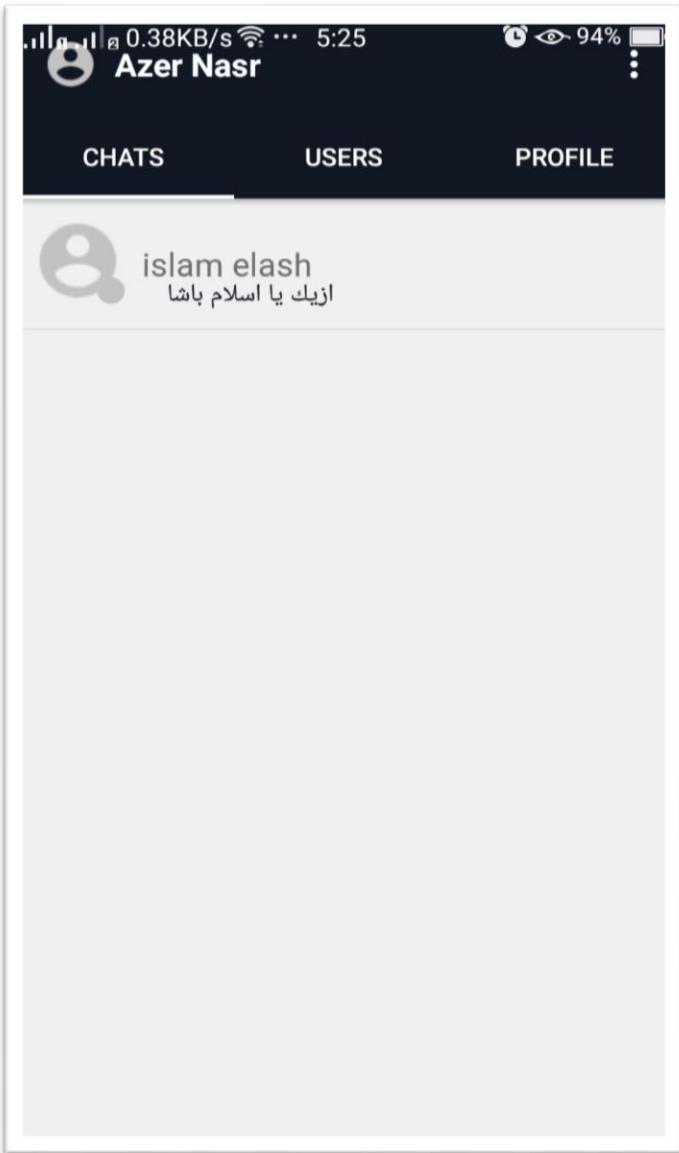


Smart Parking System

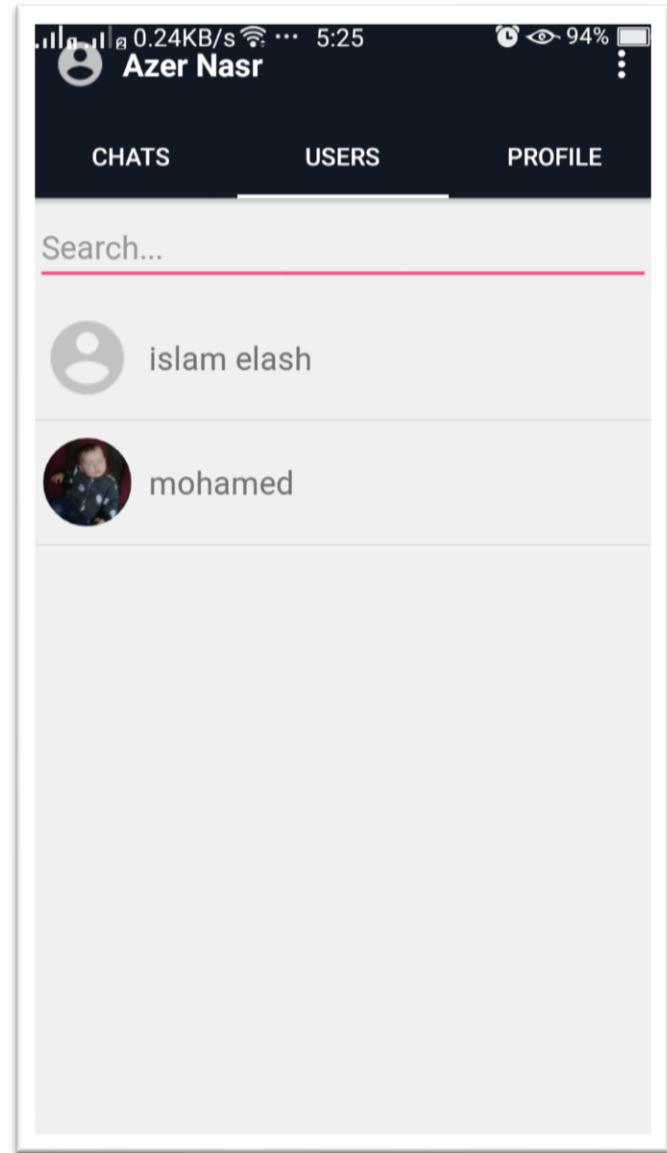
- The third screen contains profile data and car owner can update data and save it.



Smart Parking System



Chatting list

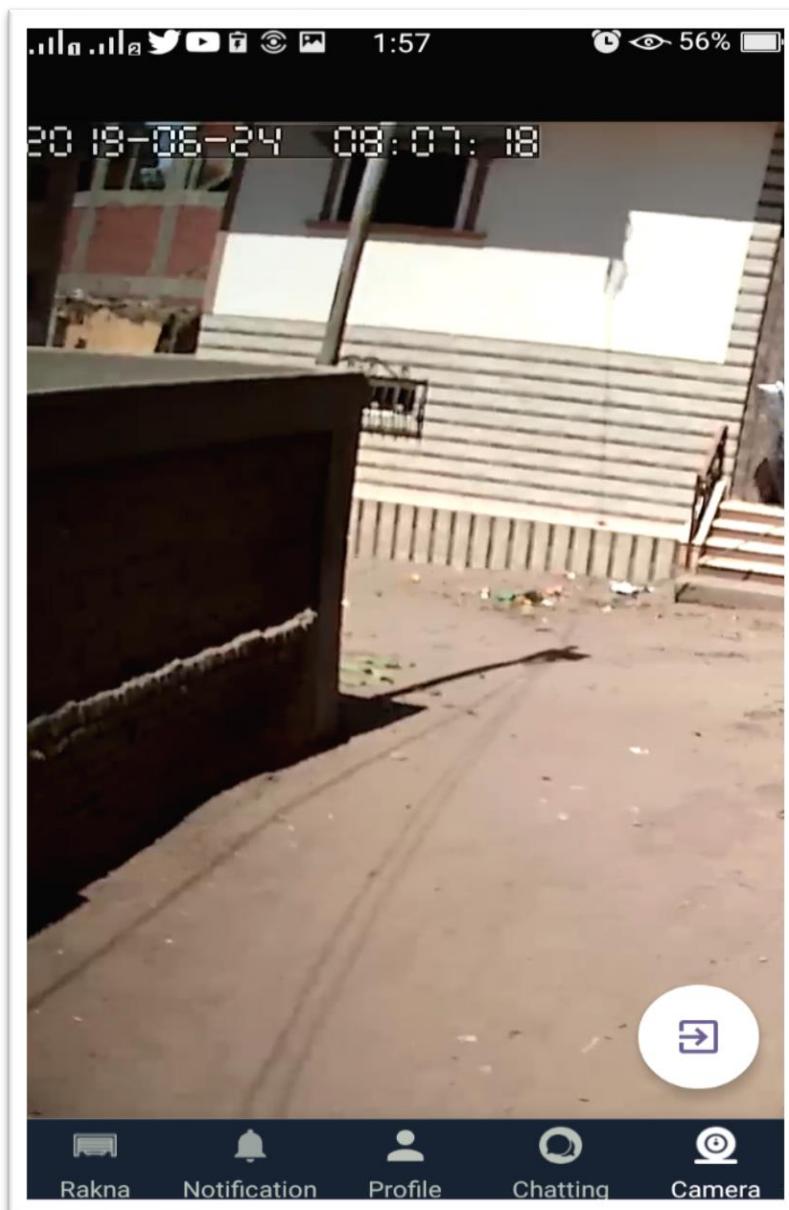


All users

Smart Parking System

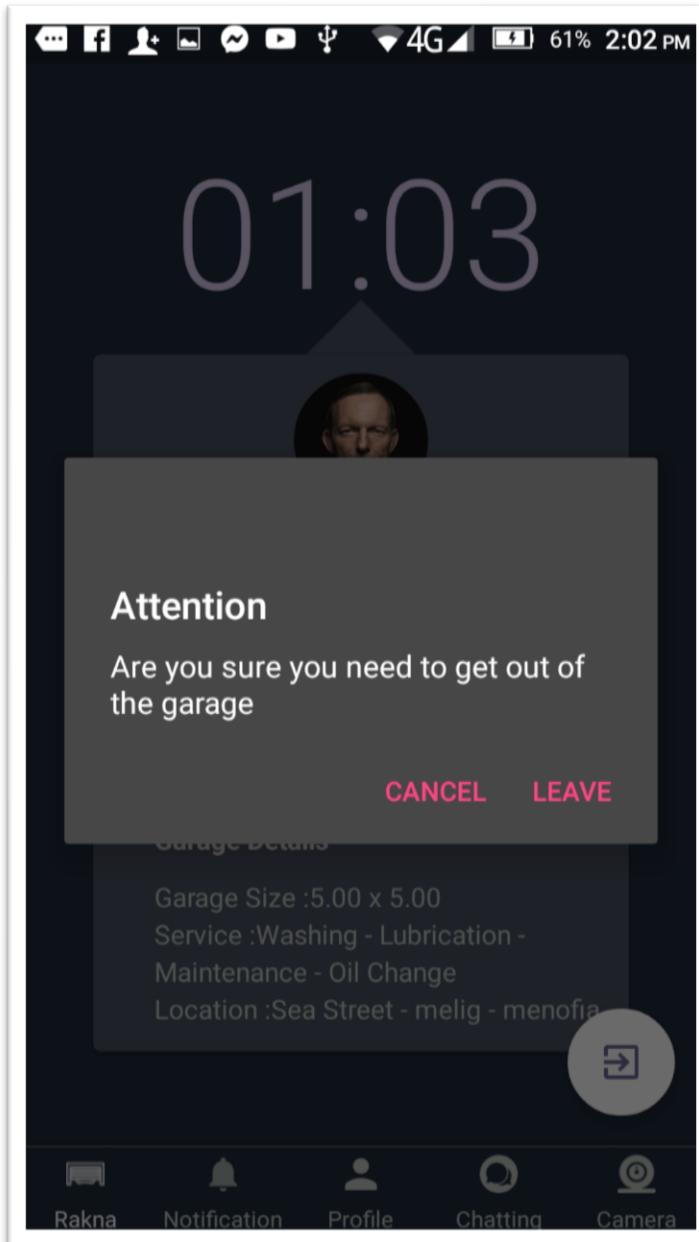
- The fifth screen contains camera

It shows automatically live stream by the system when car owner reserve garage that he can use it to check his car



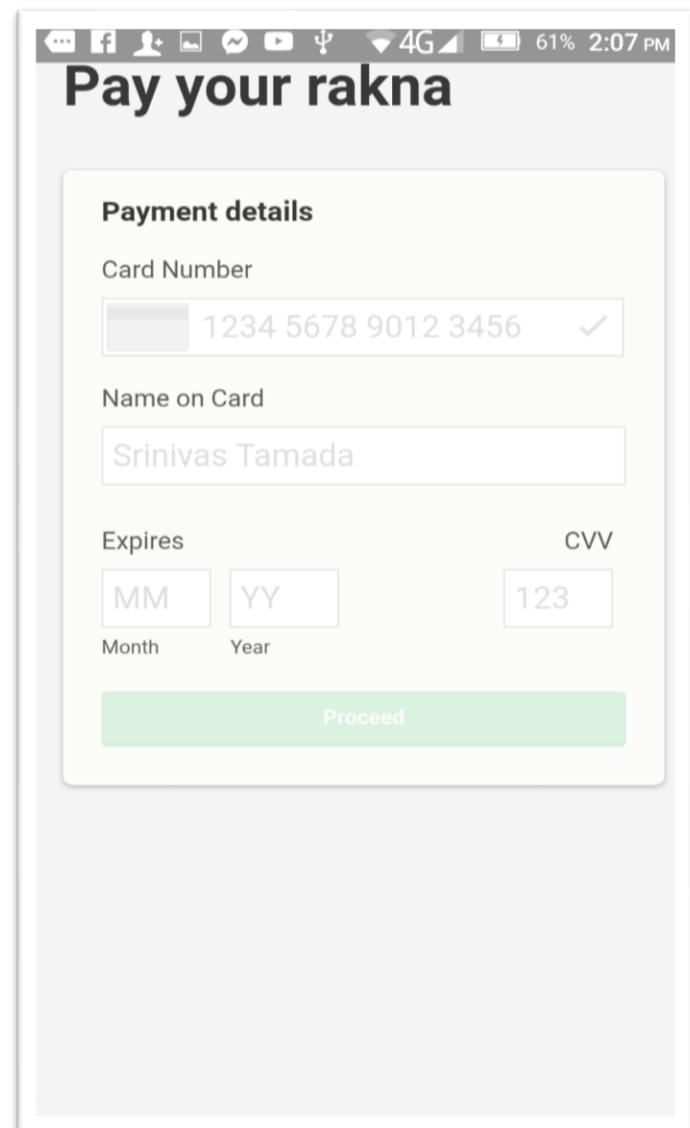
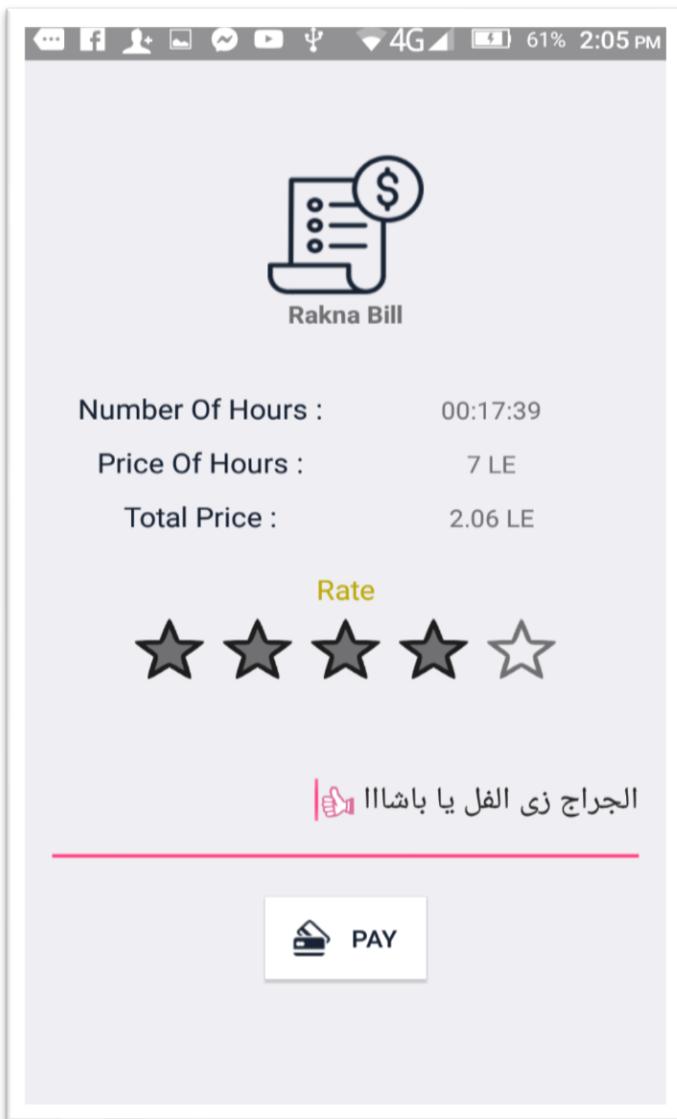
Smart Parking System

- The next is ending reservation by click the logout button

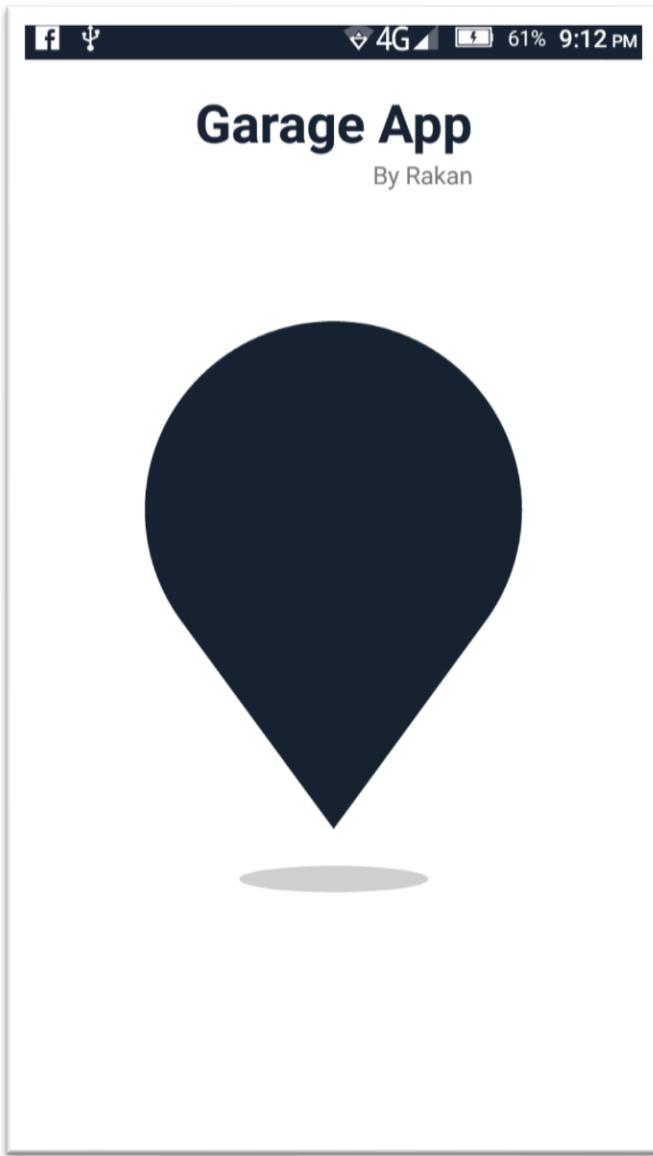


Smart Parking System

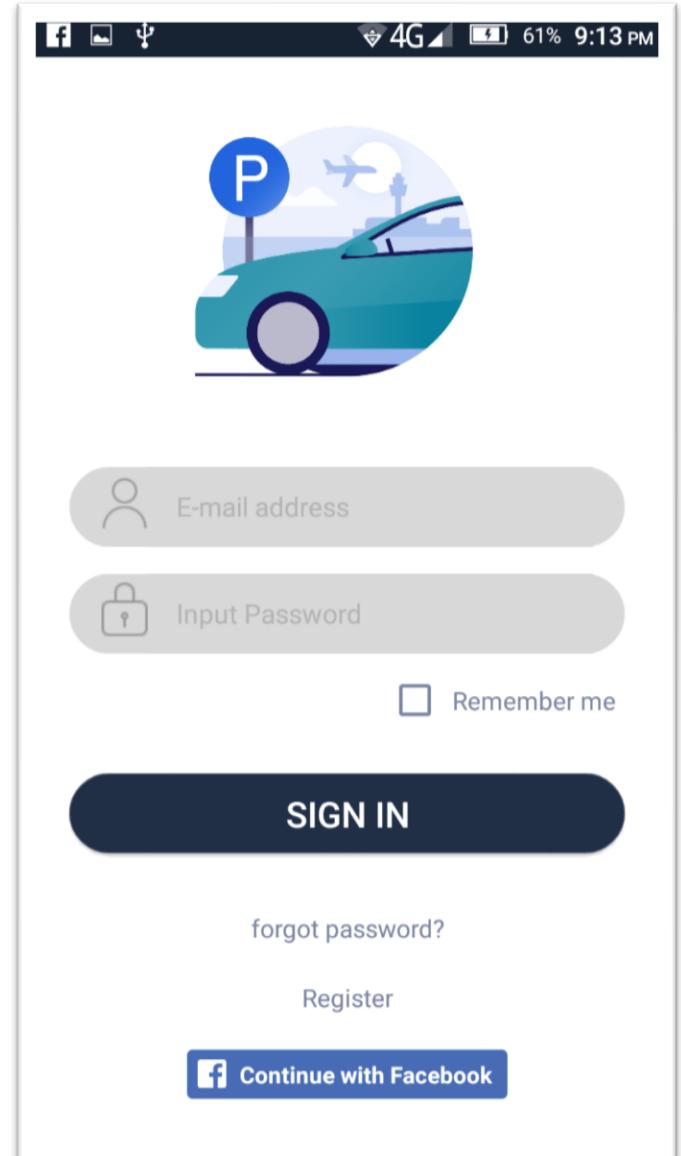
- After clicking leave button the QR Code screen will be opened to end reservation and calculate the cost of the reservation time and then open the pay screen.



The garage owner application



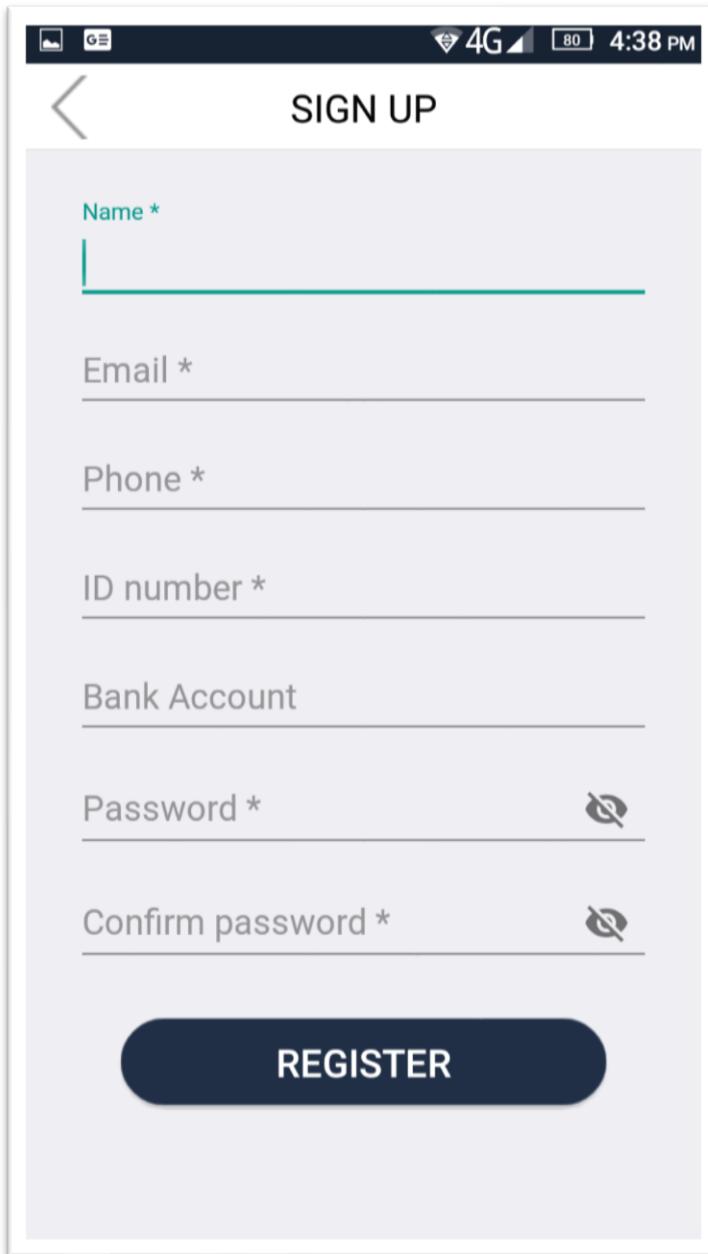
The first screen here is the Splash



Login screen

If the garage owner is registered before, he can login directly

Smart Parking System

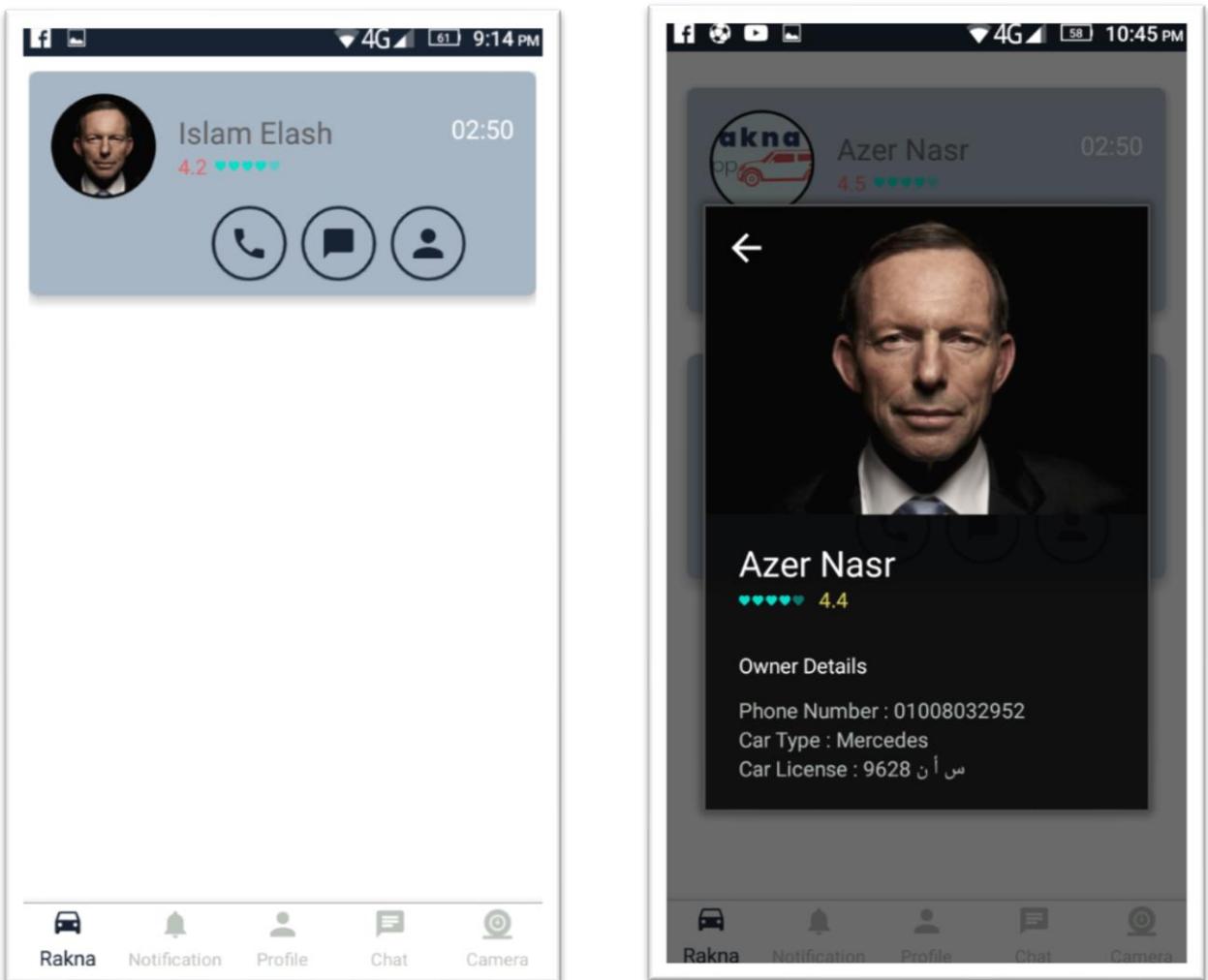


Registration Screen

If the garage owner isn't registered before, he can make quick registration by filling his personal details

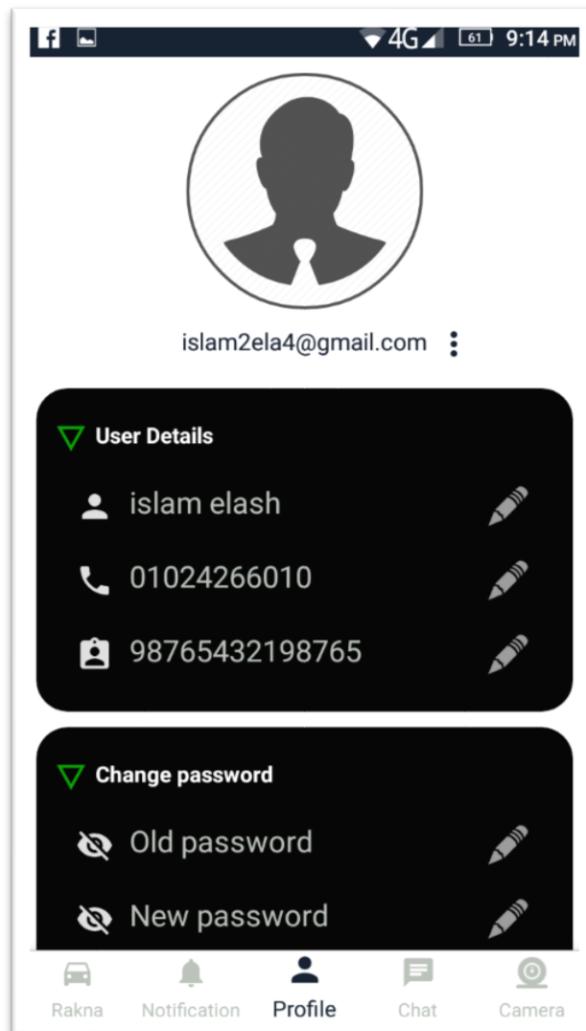
Smart Parking System

- Forget password and Facebook screens as same as car owner.
- The next is Services Screens:
 - The first screen contains online rent users that displays details about each car owner



Smart Parking System

- The second screen contains warning notifications and chatting notifications as same as the car owner.
- The third screen contains profile data and car owner can update data and save it.



Smart Parking System

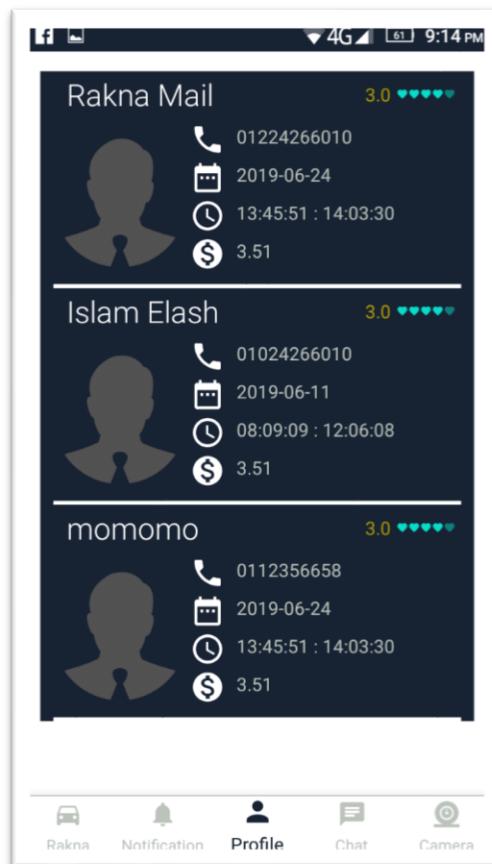
- The fourth screen contains chatting as same as car owner application

- The fifth screen contains camera

It shows automatically live stream by the system when car owner reserve garage that he can use it to check his garage as same as car owner.

- The next is history screen

It shows details about all reservations in this garage



➤ Future Plans

- Due to some controversy & collecting some ideas & reviews, the next version of the project will be somewhat different from the first version.
- The first version of the project was about a camera for view live stream to car and garage owners.
Meaning that It shows automatically live stream by the system when car owner reserve garage that he can use check his car
- Using a camera instead of using QR code or NFC.
Meaning that when the car arriving to the garage, the camera will take a photo That contains the car license and will save in the database to ensure that the car will be arrived.
When the car leaving the garage, the camera will take a photo that contains the car license and will save in the database to ensure that the car will be left.

Smart Parking System

- The camera that the garage owner use will be 360 degree to view the whole garage
 - The camera will notify the car owner and garage owner if any worn is closest from the car.
-
- its advantages

When the car leaving the garage, the camera will take a photo that contains the car license and will save in the database to ensure that the car will be left and this facilitate the process of reservation and logging out from the garage and this will save time to both users (car owner & garage owner).
 - While its disadvantages

It will require complex processing, powerful CPUs and complex image processing techniques.

- Dealing with the bank to ensure that the money will be transferred from car owner to garage owner.
- If the car owner wants to sell his car and its data saved in database, the car owner must delete the car data from database by clicking the button that exist in profile to allow the new buyer to register in Rakna Guide application.

- its advantages

this will achieve more security issues to ensure that the car is really belongs to actual owner of that car.

- While its disadvantages

It requires a little bit effort from the ex-car owner to delete its account and this not follow the standards of UX design.

- Snagging Parking Spaces with Mask R-CNN and Python using Deep Learning to Solve Minor Annoyances

Meaning that the input to the machine learning pipeline is a video stream from a normal webcam pointed out the window:

We'll pass each frame of video through the pipeline, one frame at a time.

The first step in the pipeline is to detect all possible parking spaces in a frame of video. Obviously, we need to know which parts of the image are parking spaces before we can detect which parking spaces are unoccupied.

The second step is to detect all the cars in each frame of video. This will let us track the movement of each car from frame to frame.

The third step is to determine which of the parking spaces are currently occupied by cars and which aren't. This requires combining the results of the first and second steps.

So “ **Smart Parking System** ” or **RAKNA GUIDE** version 1.1 will be more focused on security issues and using deep learning to manage cars inside garages without need of a person to organize the cars inside garages.

And will have more new features such as we enable garage owners to provide Engineering Drawings of a garage to draw 3D model for each garage to guide the user (car owner) to the place dedicated to parking its car in it by drawing a route to guide the user to the specified place to its car.