

Blockchain and Docker Assisted Secure Automated Parking Garage System

Group 4:

Luan Tran - lmt185

Khanh Nguyen - ktn31

Shalini Choudhury - sc1822

Tan Ngo - ttn64

Duc Nguyen - dhn25

Zhouyang xiao - zx150

Nainil Patel - npp71

Project blog : <https://sites.google.com/scarletmail.rutgers.edu/ruparking/main?authuser=1>

Table of Contents

| | |
|---|----|
| 1. Statement of Work & Requirements | |
| 1.1 Problem Statement | 3 |
| 1.2 Proposed Solution..... | 3 |
| 1.3 Novelty..... | 6 |
| 1.4 Glossary of Terms..... | 8 |
| 2. Functional Requirements & user Interface | |
| 2.1 Functional Requirements | 10 |
| 2.2 Non-Functional Requirements..... | 12 |
| 2.3 User-interface Requirements | 13 |
| 2.4 Stakeholders..... | 14 |
| 2.5 Actors and Goals..... | 15 |
| 2.6 Use Cases..... | 16 |
| 2.6.1 Casual Description..... | 16 |
| 2.6.2 Use cases diagram..... | 17 |
| 2.6.3 Traceability matrix..... | 18 |
| 3. User interface Specification | |
| 3.1 Sign up..... | 23 |
| A. Preliminary Design..... | 23 |
| 3.1.1 Initial Design..... | 23 |
| 3.1.2 Programming Design..... | 24 |
| B. Effort Estimation..... | 24 |
| 3.2 Sign in..... | 25 |
| A. Preliminary Design..... | 25 |
| 3.1.1 Initial Design..... | 25 |
| 3.1.2 Programming Design..... | 26 |
| B. Effort Estimation..... | 26 |
| 3.2 Reservation..... | 27 |
| 4. Domain Model Analysis | |
| 4.1 Domain Model..... | 28 |
| 4.2 Domain Model Derivation..... | 29 |
| 4.2.1 Concept Definitions (D-doing; K-knowing; N-neither)..... | 31 |
| 4.2.2 Association Definitions..... | 32 |
| 4.2.3 Attribute Definitions..... | 33 |
| 4.3 Traceability Matrix..... | 36 |
| 4.4 Mathematical Modeling..... | 36 |
| 5. Project size estimation..... | 39 |
| 6. Plan of Work | |
| 6.1 Gantt Chart..... | 40 |

| | |
|---------------------------------|----|
| 6.2 Task Phase Description..... | 40 |
| 6.3 Product Ownership..... | 42 |
| 7. References..... | 43 |

| NAME | CONTRIBUTION | PERCENTAGE |
|--------------------|---|------------|
| LUAN MINH TRAN | <u>1.Proposal:</u> - Features: User Interface - Timeline <u>2.Report:</u> - Project blog - User Interface Specification - Product Ownership - Review and fix some mistakes in the report. | 14.3% |
| HUU DUC NGUYEN | <u>2.Report:</u> - Product Ownership - Building User Interface - Actors and Goals diagram - Use case Diagram - Use case description - Sequence Diagram | 14.3% |
| TRAN TRONG TAN NGO | <u>1.Proposal:</u> - Blockchain for Customer Data Protection -Timeline <u>2.Report:</u> - Product Ownership - Proposed solution - Novelty - Actors and Goals - Use case Diagram - Fully Dressed Description & Sequence Diagram | 14.3% |
| KHANH NGUYEN | <u>1.Proposal:</u> - Problem Statements Proposed Solution and Plan of word regarding Blacklist car detection - Timeline <u>2.Report:</u> - Novelty | 14.3% |

| | | |
|-------------------|---|-------|
| | <ul style="list-style-type: none"> - Functional requirements - Non-functional requirements - User-interface requirements - Actors and Goals - Use cases and description - Domain Model - Traceability Matrix #2 - Product ownership | |
| NAINIL PATEL | <p><u>1.Proposal :</u></p> <ul style="list-style-type: none"> - Proposed Solution for Blockchain - Plan of Work for Blockchain <p><u>2.Report :</u></p> <ul style="list-style-type: none"> - Problem Statement - Traceability Matrix #1 - Mathematical Modeling - Product Ownership | 14.3% |
| SHALINI CHOUDHURY | <p><u>1.Proposal</u></p> <ul style="list-style-type: none"> - Project Description - Problem diagnosis - Proposed solution with docker - Plan of work for docker <p><u>2.Report 1</u></p> <ul style="list-style-type: none"> - Product Ownership - Docker Detailing - Proposed Solution - Use Cases and Description - Gantt Chart - Task Phase Description - Document merging - Final Review - Plan of Work | 14.3% |
| ZHOUYANG XIAO | <p><u>1. Proposal:</u></p> <ul style="list-style-type: none"> - Features: User Interface <p><u>2. Report:</u></p> <ul style="list-style-type: none"> - Product Ownership - Proposed Solution - Functional Requirements | 14.3% |

| | | |
|--|---|--|
| | <ul style="list-style-type: none">- Use Cases and Description- Domain Model Derivation- Project Size Estimation | |
|--|---|--|

1.Statement of Work & Requirements

1.1 Problem statement

Parking lots, garages and on-street parking spaces are often sources of anxiety for drivers, and currently there are not many ways to address these problems. Parking garages either use an attendant or a toll gate before entering a lot, making the process of entry and leaving the parking lot extremely slow and cumbersome, assuming that the parking lot even has space available; if it doesn't, then the driver has to back out, in potentially high traffic, causing undue stress. If the driver happens to get in to the lot, he has to take the extra time to find a parking spot and even if the customer succeed in finding one it might not be suitable for the type of vehicle he wants to garage. Another issue with parking is that the driver has no idea as to how much time he has remaining on his meter and usually overshoots or undershoots his target time, resulting in either a ticket or a waste of money spent on extra time. Finally, security is a major concern for all the customers since according to the federal Bureau of Justice Statistics, more than 11 percent of U.S. property crimes occur in parking garages and lots. Hence the parking garages are not secure and also the personal information of the customers i.e the credential and payment details may be compromised.

To address the aforementioned problems this work proposes blockchain and docker assisted secure automated parking garage system.

1.2 Proposed solution

With the aim to provide our customers with a streamlined process of easy and flexible parking. This project would implement automated garaging system to enable customers to view and reserve suitable spots for parking their vehicles. Customers through the user interface hosted on our website will be able to reserve a parking spot from the time of check-in and then extend the timing if required from the web interface by login into their account. The same slot will be updated in the parking spot availability list as soon as the customer clicks on "*spot pass on*" tab on the website and leaves the spot. This work creates an intuitive way for users to quickly locate available parking spots in a busy community, while also incentivizing users to only park during their designated time and not use the buffer period before being charged extra.

This project is backed up by multiple novel ideas which brings in great assistance to both the customer and the parking garage manager. The new customers are provisioned with one time registration with their credentials and hence, the returning customers will have the benefit of just signing in, providing vehicle type and reserving slots according to their convenience (choose from the available vacant slots). Also, depending on the size of the customer's vehicle, the user interface will provide suitable slots to choose from. In this work we will be detecting black listed cars from the amber alerts that is sent out by law-enforcement agencies, together with broadcasters, and transportation agencies to activate an urgent bulletin for crime related cases. Our detection algorithm would match all incoming cars with the stored information of the black listed vehicles and immediately send push update to the garage manager through the management user interface such that further action can be initiated. Through the push update technology, the information will be sent to the management even without querying the server for any updates. This feature will provide promptness in reacting to crime related situations. Alongside our license plate recognition system is equipped with the accuracy level of the license plate number detection, which means the image frames from the video are processed multiple times until the algorithm reaches the confidence level of 85% and above to ensure correctness of the information. This accuracy in detection is very important to verify whether the plates have been stolen, the registration status, and any notifications of the vehicle being reported stolen.

Besides using license plate scanning, it is also possible to use the ETC (Electronic Toll Collection) system. Through the special short-range communication between the on-board electronic tag installed on the windshield of the vehicle and the microwave antenna on the ETC entrance of the toll gate, the computer network technology is used to perform background settlement processing with the bank. In this way, the no-stop parking charge would be achieved. More importantly, the customers could also recharge their ETC cards from the bank transfer or by going to the given top-up sites. So that there is no need for the customers to link their bank card or credit card on another website, which can definitely reduce the hazard for the personal information leaked or property stolen.

Another novel idea that has been included in this project is blockchain assisted customer data protection. Data privacy protection concerns are driving new regulations around the world. As they act to protect data privacy online and improve personal data protection, our mechanism of data protection would go beyond just complying with all the new requirements will build trust with consumers and users and stand out from other competitors in the business. Since encryption is central to block chain, a smart blockchain based application will improve data integrity and will have no single point of failure since the data is distributed and decentralized. This brings in a feature where a consumer has the right to instruct a business not to sell personal information to a third party, meaning that a business that tries to sell a blockchain network will have a harder time removing individual blocks from each chain.

The docker based containerization of information is another feature which adds novelty to this work. This method of decentralized information exchange will act as a backbone to support the infrastructure of automated parking garage. In order to maintain automation in parking and eliminating any manual assistance, a system has to be built where data/information access and exchange has to be done without complicated and cumbersome means and with robust support. Some parking companies like Icon Quik Park own 300 garages around New York City, require infrastructure which would enable scalability by providing adequate data management and exchange and also a platform for hosting applications. To comply with this requirement for adequate management of available parking spots and combined pricing across multiple garaging location, containerized information exchange between multiple garaging locations is executed. This method of data exchange is real-time and for better assisting the customers the quick information is available regarding in which garaging location more spots or preferred spots are available as well as pricing in different location. Also if a customer decides to use different garage locations under the same management throughout the day for their convenience, then combined pricing will be charged once when the customer finally checks out for the day. This provides great flexibility to the customer since there will be only one time charge and no recurrent payment every time the customer checks in to a garage location managed by the same owner. The combined pricing will be charged depending upon the parking rate in a given location at a particular time of the day, hence this is dependent on the dynamic pricing of a garage location.

As can be seen from the above discussion that to coordinate availability of parking, pricing and exchange of crime alerts and black listed car information, between different garaging location it is extensively important to host these data/state in a lightweight virtual software entity which can be exchanged between different garaging locations further apart such that all the garaging related information are available at all the same owner administered parking facility. All the parking management applications and its dependencies will also be supported in the container. Containers, also are generally less resource-demanding and faster to instantiate at the destination location that it is offloaded to.

Fully automated parking service theoretically eliminate the need for parking attendants. In addition to this, it is only necessary to register at the first time and then the vehicles' information of the customers will be automatically recorded, which makes the entry and exit of parking more rapid. At the same time, it has also a more reasonable toll system and more convenient search system of parking spots, and can deal with sudden cases like the amber alerts. So, as long as the customer's privacy is guaranteed, this will be a comprehensive upgrade compared to the traditional parking.

1.3 Novelty:

We investigated the previous reports including the Fall 2018, Spring 2013 and Spring 2012 in order to distinguish the previous works with our renovated work. We will provide a summarization features of previous works and compare our work with them. In the following is the features listed in the report of Fall 2018:

1. Website Registration
2. Online website for reservation
3. Customers are supported to pay bills and exit garage by security guards.
4. Alarm user if they are different
5. Alarm not working-sensor to authority
6. Garage remodeled such that an elevator can lift vehicles to different decks
7. Single operation (exiting and entering) to the garage at one time
8. Entering or exiting of a vehicle is tracked by camera-base plate recognition software
9. Elevator works only when license plate is verified
10. Customers will receive confirmation code via email or SMS to enter the parking lot
11. When a customer is in the expected spot, a message will be sent to them.
12. Non-reservation customers will be notified when parking lots are available
13. Pay the reservation through mobile
14. Floor sensors for detecting occupancy and monitoring traffic on the floor
15. Cars are lifted up to the right spots by elevators
16. Dynamic price based on conditions
17. Owner can keep track of right spot through monitor
18. Extend their time parking when the slot is available.

Moreover, for the 2013's features list:

1. The system provides three kinds of account, namely: customer account, manager account, employee account.
2. It brings cars to a checkpoint and take pictures of the car before it enters the garage and also when it is about to leave for security reasons. A picture of a car is captured when it enters and exits the garage.
3. Frequent customers are memorised through reading plates
4. It integrates a simulator to test the system
5. It deals with overstate problem.
6. Customer's data is encrypted

7. Daily report about account information or parking data is saved once a day

Additionally, for the 2012's features listed:

1. Support confirmed reservation (no credit card) and guaranteed reservation (with credit card)
2. If customers do not show up within 1 hour, the reservation spot is free for another vehicle
3. Rented and borrowed transportation are allowed to park.

By comparing with the aforementioned features in the previous reports, the main novelty of this project will be listed in the following:

1. **Blockchain for Customer Data Protection:** Blockchain is used to improve data integrity and security. It ensures that your data is encrypted, which means that modification in data is a difficult task, hence this would give customers that their personal and payment information are safe.
2. **Detection of blacklisted cars:** Our system will detect the blacklisted car by detecting the blacklisted license plates which are circulated by the amber alerts or are in the crime hotlist. This would allow detection and reduction of crime rates and add to the security.
3. **Docker/ Containerized Information Exchange:** This method is an improvised idea of information exchange between various parking locations. With this method real-time information exchange regarding combined pricing, parking slot availability and crime alerts can be done.
4. **User Interface:** The aim is to create user-friendly interface to provide easy navigation to various features such as sign-up, sign-in, reservation, cancel reservation, see availability of parking spots according to vehicle sizes, offers/discounts, dynamic price and time elapsed.
5. **Slot allotment depending on vehicle size:** The garage is provisioned with slots for different vehicle sizes. Depending on the size of the vehicle the customer wants to garage, the user interface will give list of available slots to choose from.
6. **Combined Parking:** A features that can help the customer park multiple times at multiple locations and will be charged at the end of the day. For example, when the customer's lot starts at 2:00PM and ends at 3:00 P.M in New Jersey , they want to park at another garage location in New Jersey from 3:00 P.M to 4:00P.M and at the end of 4:00 P.M, the charge will be summed of 2:00PM-3:00PM and 3:00P.M to 4:00P.M and be charged at once when the customer leaves at 4:00 P.M.

1.4 Glossary of Terms:

Blacklisted plates: The plates which are announced by the law enforcement for being involved in any crime.

Online Reservations: Reserved parking spots by visiting the website.

Registration: The customer sign-up with their credentials and these information are available in the company database. When registering before reserving, the customer is asked to input their name, genders, phone number/email, credit card number, address, driver's license number and date of birth. By registering, it allows the customer to reserve parking spots.

ETC (Electronic Toll Collection): It is the system aims to avoid the delay on toll roads, HOV lanes, toll bridges and toll tunnels by collecting tolls without cash and without requiring cars to stop.

Manager's account: It is the account which the admin of the parking garage has access to. The manager account keeps track of the dynamic pricing, customer details, slots availability. The manager account have access to information related to other parking locations owned by the same company.

Confirmed Reservations: Reserved single use parking spots that are paid for ahead of time. Done by visiting the website.

Customer: Person who enters the garage with either a parking reservation or looking to make a walk-in reservation.

Database: Hosted on the website, used to store customer data and parking information.

License plate scanning: is going to be a scanner at a checkpoint that will record the license plate number of the vehicle and store it into the systems database.

Extension: Customers are allowed to increase the amount of time their car is in the garage, as long as there is available time slots once their time is over.

Buffer time: The extra time to allow the customer to check out from the spot allotted to him

No-Show: The act of missing a reservation. The manager will still collect the payment due to paying upon reservation.

Overstay: An unavailable spot that may impede future reservations due to the customer not returning within the allotted reserved time.

Understay: A newly vacant spot due to a customer leaving before the conclusion of their reserved time.

Walk-ins: Parking customers who walked into the garage without a reservation.

Servers: Server is a device that provides functionality for other computer programs or devices.

Containerization: Containerization involves bundling an application and data together with all of its related configuration files, libraries and dependencies required for it to run in an efficient way across different computing environments.

Decentralized: In a decentralised system lower level components operate on local information to accomplish global goals.

User Interface: The user interface (UI) is the point of human-computer interaction and communication in a device.

Encryption: In cryptography, encryption is the process of encoding a message or information in such a way that only authorized parties can access it.

2. System Requirements and Functional Requirements

2.1 Functional Requirements

| Req | PW | Description |
|------------|-----------|--|
| REQ-01 | 6 | The system shall create accounts for customers. |
| REQ-02 | 6 | The system shall allow the customer to log in |
| REQ-03 | 6 | The system shall allow customers to view and reserve their suitable parking spots. |
| REQ-04 | 7 | The system shall allow walk-in customers to reserve in parking toll gate. |
| REQ-05 | 6 | The system shall allow the reservation to be canceled before the agreed time. |
| REQ-06 | 8 | The system shall allow the customer to change the time reservation based on their demands |
| REQ-07 | 8 | The system shall allow the customer to edit their information |
| REQ-08 | 5 | The system shall allow the customer to see the past transactions. |
| REQ-09 | 6 | The system shall show the remaining time to customers. |
| REQ-10 | 7 | The system shall allow parking time extension on the website. |
| REQ-11 | 5 | The system shall update the parking slot availability list as soon as the last customer leaves the parking spot. |
| REQ-12 | 5 | The system shall update the parking slot availability list as soon as one customer reserves the parking spot. |
| REQ-13 | 9 | The system shall provide suitable parking spots based on the size of the customers' vehicles. |
| REQ-14 | 5 | The system shall provide customers with a road map to go to the reserved parking spot. |
| REQ-15 | 7 | The system shall scan license plates. |

| | | |
|--------|---|---|
| REQ-16 | 8 | The system shall recognize registered customers via plate number. |
| REQ-17 | 7 | The system shall detect the frequent of customers via plate number and address the type of members based on the frequently use of the customer. |
| REQ-18 | 6 | The system shall update the blacklisted car from the website every time. |
| REQ-19 | 7 | The system shall detect blacklisted cars from the amber alerts for all incoming cars. |
| REQ-20 | 9 | The system shall exchange the data between multiple garaging locations regarding the customers and blacklisted car |
| REQ-21 | 5 | The system shall inform the manager about the blacklist-car warning. |
| REQ-22 | 4 | The system shall detect the accuracy level of the license plate number. |
| REQ-23 | 7 | The system shall make a combined pricing across multiple garaging locations |
| REQ-24 | 6 | The system shall provide real-time dynamic pricing of garaging locations. |
| REQ-25 | 7 | The system shall provide a reasonable charging standard on the basis of the demand. |
| REQ-26 | 5 | The system shall allow the customer to have 5 extra minutes to take the car before being charged by the next time slot. |
| REQ-27 | 8 | The system shall allow payment on the website. |
| REQ-28 | 7 | The system shall allow payment in the parking toll gate. (by cash/bank card/QR code) |
| REQ-29 | 6 | The system shall allow the changes of payment methods. |
| REQ-30 | 5 | The system shall mail or message customers in case of an emergency. |
| REQ-31 | 9 | The system shall protect the personal information of customers. |
| REQ-32 | 7 | The system shall protect the property of customers. |
| REQ-33 | 5 | The system shall have the capable of managing customer information and availability of parking slots. |
| REQ-34 | 7 | The system will ask if their are another reservation under the same user at the same time in order to make sure and reduce the cost. |

| | | |
|--------|---|---|
| REQ-35 | 6 | The user data or information will be input to the database and stored in the main database. |
| REQ-36 | 4 | The system will automatically charge for the cancellation fees. |
| REQ-37 | 8 | The system will connect all the garages that apply our system together through the main database. |
| REQ-38 | 5 | The system will show the manager the available slot in order to update the parking lot spaces. |
| REQ-39 | 4 | The system will be back-up and will be encrypted in order to protect the data and evade hackers. |
| REQ-40 | 6 | The system will send the QR code when customer finishes reservation. |

2.2 Non-Functional Requirements

| Req | PW | Description |
|------------|-----------|---|
| REQ-41 | 4 | The online reservation need to be done in the place with providing wifi,3G by the customer |
| REQ-42 | 7 | The customers can check the history transaction every time. |
| REQ-43 | 9 | The customers can edit their reservation or cancel the reservation before parking exactly 2 hours. After 2 hours,the reservation cannot be changed. |
| REQ-44 | 6 | When registration completes, the customers need to confirm their email |
| REQ-45 | 4 | A customer can reserve online when coming to the garage and choose the desired slot. |
| REQ-46 | 4 | Recovery time will not exceed over 5 minutes |
| REQ-47 | 8 | When the customer complete the payment, the money transaction will be transferred to the owner and based on the contract to receive the fee when applying our system |
| REQ-48 | 9 | The main database will the the one that stored all the reservation info and will transmit the data to one parking lot if the customer wants to change the parking lot. |
| REQ-49 | 7 | The system will be tested and debugged in order to check for the potential errors might have. |
| REQ-49 | 9 | The system will be simulated in order to check the license plate reader and check if the customer can register and the data information will be transmitted to the database with the notification system. |
| REQ-50 | 6 | The system will not lose the reservation online with being updated every time. |
| REQ-51 | 7 | If the customer's car is being lost in the manager's garage, our system and the manager will provide an insurance for the car and pay full compensate for it. |

2.3 User-interface requirement

| Req | PW | Description |
|------------|-----------|--|
| REQ-52 | 5 | In order to create an account, customers need to have a mobile device or state-of-the-art technologies that could connect to wifi or 3G and connect on the websites, create an account.. When registering the account, customers need to input their information such as last name, first name, email, mobile number,... |
| REQ-53 | 8 | To log in to the customer's account, the customer needs to put the adequate email and password. |
| REG-54 | 5 | If the customer forgets the password, the UI will show the reset password features in the login page |
| REG-55 | 5 | If the customer hasn't registered the account on our page, the UI will show the unregistered features in the login page and lead to the signup page |
| REQ-56 | 9 | To update and edit the customer's information, the customers need to log into their account and update or edit their information |
| REQ-57 | 9 | To make an online reservation through web, customers need to log in to their account, choosing the hours which is suitable for them and choosing the slots based on their demands ; lastly,choose their payment method and make a submission. Then the system will receive the data and give back the user ID code with the confirmation hour through gmail. |
| REQ-58 | 7 | To update and edit the customer's reservation, the customers need to log into their account and update or edit or cancel their reservation (for online customer) |
| REQ-9 | 7 | To change the customer's slot or cancel their reservation, the customers need to log into their account, choose the reservation and input the confirmation ID code. Then, the users need to confirm that they want to change their slots or cancel the reservation with a box yes or no. |

2.4 Stakeholders

Stakeholders are the people interested in the success of the organization. They are classified as primary and secondary stakeholders. In our case, the primary stakeholders are the garage owner and managers, who will be directly utilizing the system to increase efficiency and profit of the organization. Another set of stakeholders will be the customers, who will also use our system to have a better and systematic garaging experience.

2.5 Actors and Goals

| Actors | Goals | Use Cases |
|------------------------|---|---|
| Parking User Interface | To display the available slots for the customer on the website and confirm their payment information in order to reserve slots in the customer-desire garage in ahead of time or cancel the reservations. | UC3, UC4, UC5, UC7, UC8, UC18, UC19, UC22, UC23 |
| Parking Interface | To display the empty slots for the walk-in customers and allow them to park when entering the garage | UC3, UC4, UC5, UC6, UC8, UC18, UC19, UC23 |
| System | Update customer's information, reservation and payment method | UC24 |
| System | Update the parking information, the time elapses. time left and available slots if the customer wants to extend through mail | UC11, UC12, UC13, UC14, U23 |
| Security System | To protect the private information of the customer | UC21 |
| Security | To ensure that customers booking the slots on the website and the coming customers are the same. | UC20 |
| Camera | Detect blacklisted car | UC17 |
| Garage Owner | To define a price for each slot and the demand with it | UC8 |
| Garage Owner | To keep track parking-slot, in the garage and how much time that the customers have left. | UC11, UC12, UC15, |
| Garage Owner | To receive the warning from the camera of blacklist-car | UC17 |

| | | |
|----------|--|-------------------------------------|
| Customer | To register and log in on the website | UC1, UC2 |
| Customer | To modify or update their information or payment method on the website | UC24,UC25 |
| Customer | Choosing paying online or when exiting the garage | UC25 |
| Customer | To check for the available slots and reserve with the adequate money for customer-demands on the internet | UC3, UC4, UC5, UC8, UC9, UC10, UC18 |
| Customer | To check for the available slots and take the slots with customer-demands regarding no-advanced booking customers. | UC3, UC4, UC5, UC6, UC18, UC19 |
| Customer | Arriving at the garage | UC15 |
| Customer | Exit the garage and pay for the parking hours. | UC9, UC10, UC16 |
| Customer | Receive their parking information through mail | UC11, UC12, UC13, UC23 |
| Customer | Cancel the reservation on the website | UC7 |
| Customer | Paying for the parking hours | UC9, UC10 |

2.6 Use Cases

2.6.1 Casual Description

1. **UC1:** Sign Up - Customer create an account on our website
2. **UC2:** Log In - Customer use their registered credential to access the website
3. **UC3:** Slot Availability - Check if slots are available for parking
4. **UC4:** Slot Preference - Select from the available parking slots as per choice
5. **UC5:** Adaptive Parking - Allotment of slots according to the size of the vehicle
6. **UC6:** Walk-In - Customers without advanced reservations
7. **UC7:** Reservation Cancellation Or Editing - Cancel reservation before showing up
8. **UC8:** Dynamic Pricing - Charging customer depending on the time of the day when the demand is high/low
9. **UC9:** Payment - Charge your customers at the end of parking tenure
10. **UC10:** Combined Payment - Let customers use multiple parking locations and be charged once finally when he leaves.
11. **UC11:** Time Elapse - To keep an account of time elapsed since parking.
12. **UC12:** Time Update - Send customer update about the time left before reservation expiry
13. **UC13:** Extra Parking Time - Bonus time allotted to the customer before they are charged for the next time slot
14. **UC14:** Reservation Extension - Extend current reservation time from website
15. **UC15:** Arrival - Customer entering the garage
16. **UC16:** Departure - Customer exit the garage
17. **UC17:** Blacklisted Car Alert - Inform garage manager about blacklisted car detection
18. **UC18:** Suitable Garage Location - Choose the preferable parking location
19. **UC19:** Comparative Price Parking - Park at the location which offers least price
20. **UC20:** Parking Security - Send code to the customer when making online reservations or offline, when the customer gets to the desired slot, to get into the slots, the customer needs to give the code to the system and then the system will allow the customer to get in.
21. **UC21:** Information Privacy - Encrypt customer information
22. **UC22:** Email Updates - Send pay amount, security code, time elapsed and start of buffer time.
23. **UC23:** Reservation - To reserve a slot in 2 hours

| Actors | Use case | Description | REQ |
|---------------|---|--|------------|
| Customer | UC-1: Signup | Customer creates an account on our website | |
| Customer | UC-2: Log in | Customer uses their registered credential to access the website | |
| Customer | UC-3: Edit information | Customer can edit their information on the UI | |
| Customer | UC-4: Reset password | Customer forgets password will have a link which sends to their email can reset their password | |
| Customer | UC-5: Walk-in | Customers without advanced reservations | |
| Customer | UC-6: Suitable Garage Location | Choose the preferable parking location | |
| Customer | UC-7: Slot Availability | Check if slots are available for parking | |
| Customer | UC-8: Slot Preference | Select from the available parking slots as per choice | |
| Customer | UC-9: Comparative Price Parking | Park at the location which offers least price | |
| Customer | UC-10: Reservation | To reserve a slot in 2 hours | |
| Customer | UC-11: Reservation Cancellation Or Editing | Cancel or Edit the reservation before showing up | |
| Customer | UC-12: Arrival | Customer enters the garage | |
| Customer | UC-13: Departure | Customer exits the garage | |

| | | | |
|----------------|-------------------------------------|---|--|
| Customer | UC-14: Combined Payment | Let customers use multiple parking locations and will be charged once finally when he leaves. | |
| Customer | UC-15: Reservation Extension | Extend current reservation time from website | |
| User Interface | UC-16: Show reservation | The UI will show the confirmation regarding the reservation. | |
| System | UC-17: Blacklisted Car Alert | Inform garage manager about blacklisted car detection | |
| System | UC-18: Email Updates | Send pay amount, security code, time elapsed and start of buffer time. | |
| System | UC-19: Adaptive Parking | Allotment of slots according to the size of the vehicle | |
| System | UC-20: Time elapse | To keep an account of time elapsed since parking. | |
| System | UC-21: Time update | Send customer update about the time left before reservation expiry | |
| System | UC-22: Extra Parking Time | Bonus time allotted to the customer before they are charged for the next time slot | |
| System | UC-23: Payment | Charge your customers at the end of parking tenure | |
| System | UC-24: Dynamic Pricing | Charging customer depending on the time of the day when the demand is high/low | |

2.6.3 Traceability Matrix

| | CASE | Register Online | Log-In Online | Make Reservation Online | Walk-in | Enter | Depart |
|-------------------------------|------|-----------------|---------------|-------------------------|---------|-------|--------|
| REQUIREMENT | | | | | | | |
| Create a New Account | | X | | | X | | |
| View or Reserve Parking Spots | | | X | | X | | |
| Change or Cancel Reservations | | | X | | | | |
| Update Spot Availability | | | | X | X | | |
| Provide Real-Time Pricing | | | | X | X | | |
| Allow Payment | | | | X | X | | |
| Scan License Plates | | | | | | X | X |
| Recognize Customers | | | | | | X | X |

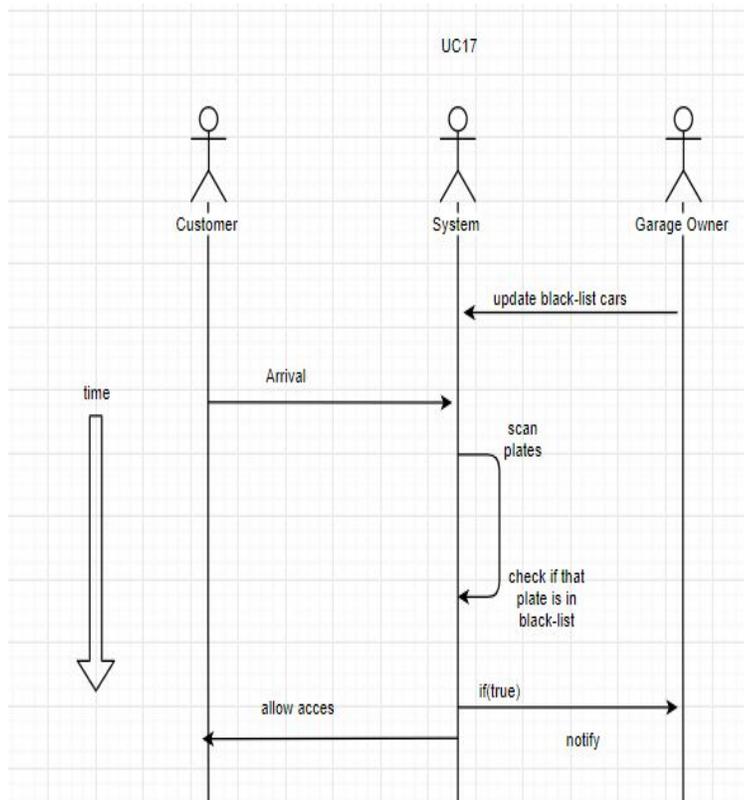
| | | | | | | | |
|-------------------------------|--|---|---|---|---|---|---|
| Detect Blacklisted Cars | | | | | | X | X |
| Give Internal Directions | | | | | | X | X |
| View or Extend Remaining Time | | | X | | | | |
| Send Email | | X | | X | | | X |
| Edit Profile Data | | | X | | | | |
| View Profile History | | | X | | | | |
| Share Data between Garages | | | | X | X | X | X |

2.6.4 Fully Dressed Description & Sequence Diagram

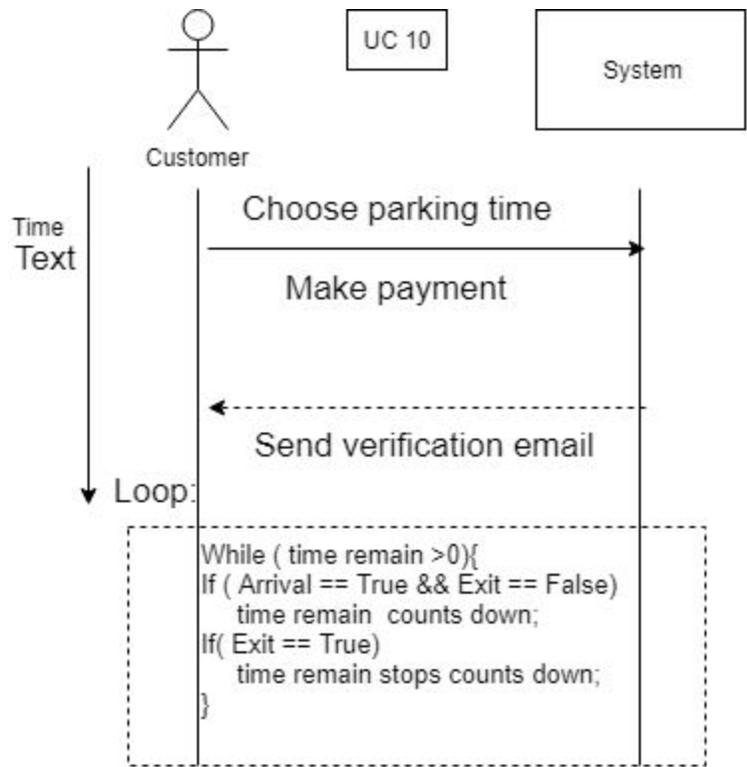
| |
|--|
| Use Case UC-17: Blacklisted Car Alert |
| Related Requirements: REQ-03, REQ-14, REQ-17, REQ-18, REQ-19, REQ-20, REQ-21 |
| Initiating Actor: Garage Owner |
| Actor's Goal: To being notify if there is a blacklisted car in his/her garage. |
| Participating Actors: Garage Owner ,Customer, System |
| Preconditions: The garage owner is able to access blacklisted car data from the law enforcement |
| Postconditions: The customer is allowed to park either the plate is in the blacklist or not. |

Flow of Events for Main Success Scenario:

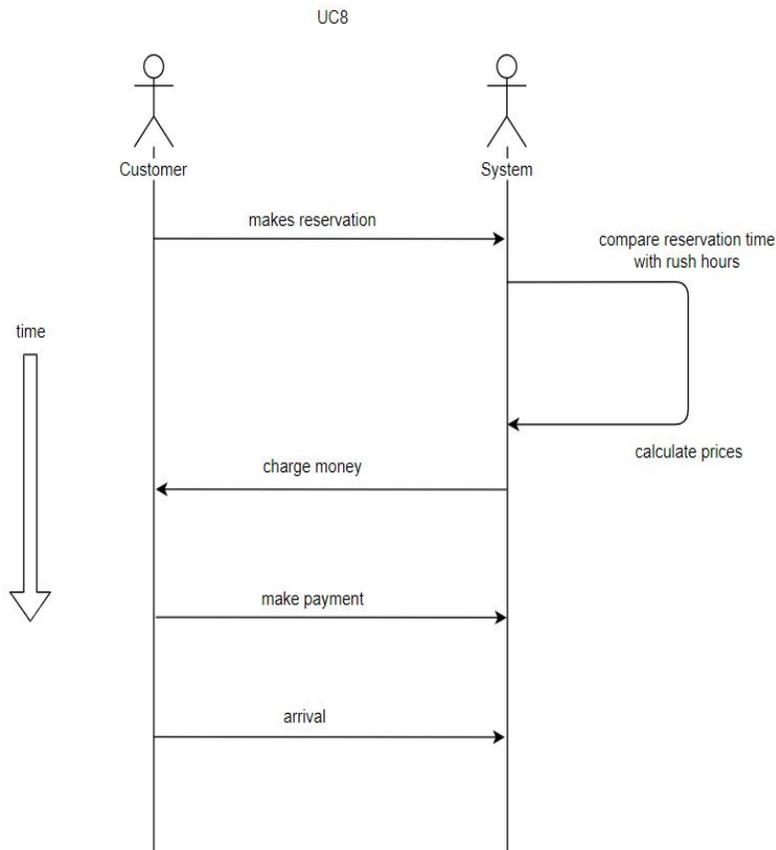
- >1. The garage owner updates the blacklist plate.
- >2. Customer accesses the garage throughout the main gate and the camera scans the plate
- <-3. The system compares the plate with the blacklisted car data
 - a. If the plate is in the blacklist, notify the garage owner.
- <-4. Allow the customer to access.



| |
|--|
| Use Case UC-10: Combined Payment |
| Related Requirements: REQ-02, REQ-08, REQ-19, REQ-22, REQ-38, REQ-39 |
| Initiating Actor: Customer |
| Actor's Goal: To have the ability to continue the parking time in other locations in the system |
| Participating Actors: Customer, System |
| Preconditions: The customer is required to have an account in the system. |
| Postconditions: The customers are required to pay more if their parking time is over. |
| Flow of Events for Main Success Scenario: ->1. The customer makes payment on our website. <-2. The system sends verification email. --3. The system enters a loop and the exit condition is when time elapses is equal to the paid parking duration . a. If the customer check-in, time elapses starts counting. b. If the customer check-out, time elapses stops counting. |



| |
|---|
| Use Case UC-8: Dynamic Pricing |
| Related Requirements: REQ-5, REQ-8, REQ-12, REQ-19, REQ-22 |
| Initiating Actor: Customer |
| Actor's Goal: To charge customers based on the time making reservation on a day. |
| Participating Actors: Customer, System |
| Preconditions: The customer is required to have an account in the system and make advanced reservation. |
| Postconditions: The customers are required to pay more if their parking time is over. |
| Flow of Events for Main Success Scenario: ->1. Customers log in to the website and make reservations. <-2. System calculates the price based on the time making reservation. ->3. Customers make payment. --4. Customers arrive at the parking lot. |



3. User interface Specification

3.1 Sign up - UC1:

A. Preliminary Design

3.1.1 Initial Design:

Sign Up

First Name Last name

Your Email

Mobile Number

Password

Confirm Password

Address

DL Number

Date of Birth

Month Day Year

Submit Now

3.1.2 Basic Programming Design:

Sign Up

First Name
Last Name
Your Email
Mobile Number
Password
Confirm Password
Address
DL Number

Date of Birth

Month Day Year

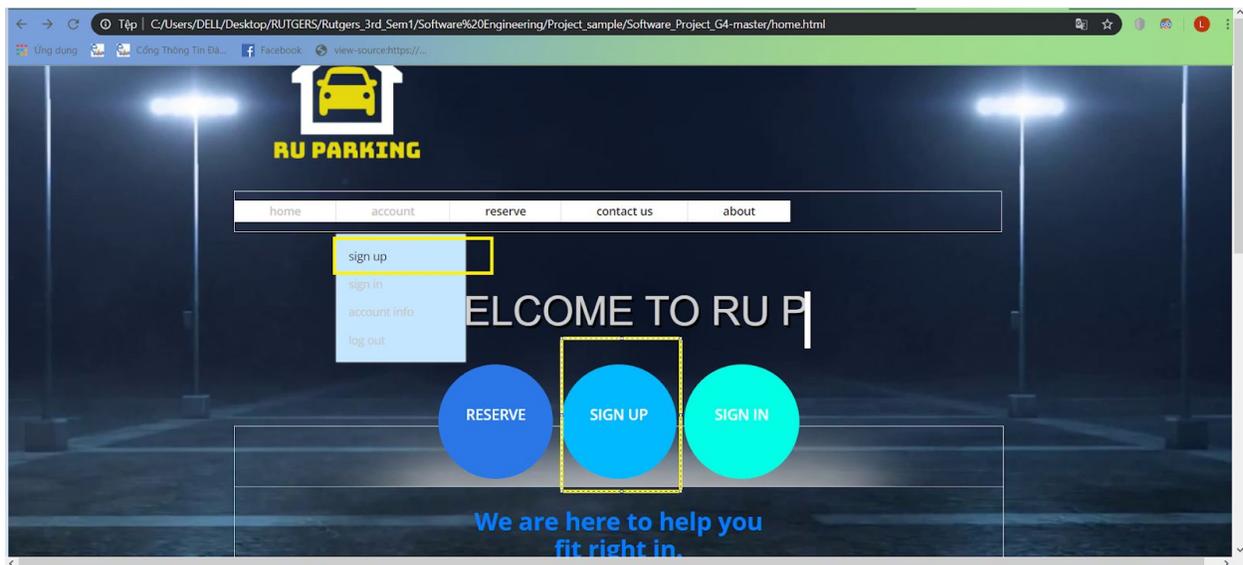
Male Female Others

I agree to the terms of use

SUBMIT

A. Effort Estimation:

Navigation: total 1 click either on the circle or on the navigation bar:



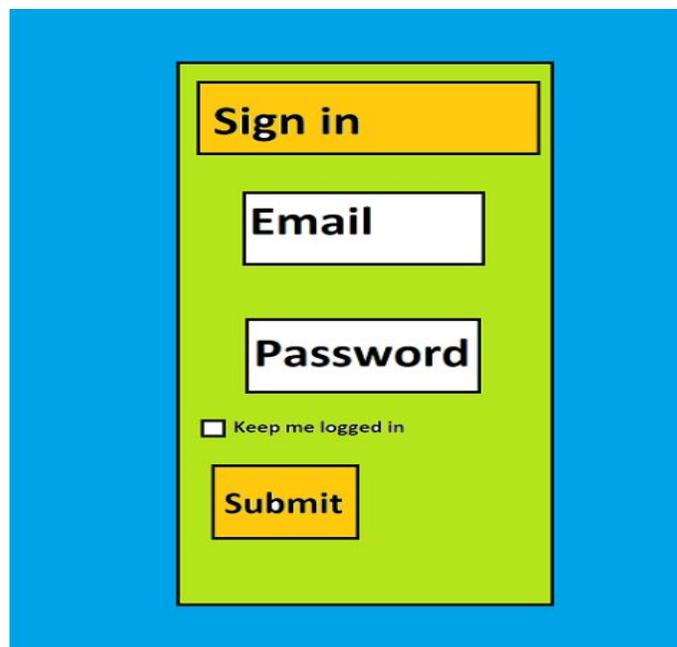
Data Entry: Total 1 click on Sign Up and 11 keystrokes as follow:

1. Click Sign Up either on the circle or navigation bar then enter information:
 - First Name
 - Last Name
 - Email (This is Username)
 - Mobile Number
 - Password
 - Confirm Password
 - Address
 - Driver License Number
 - Date of Birth
 - Gender
 - Agreement
2. Click button “Submit”

3.2 Sign in - UC2:

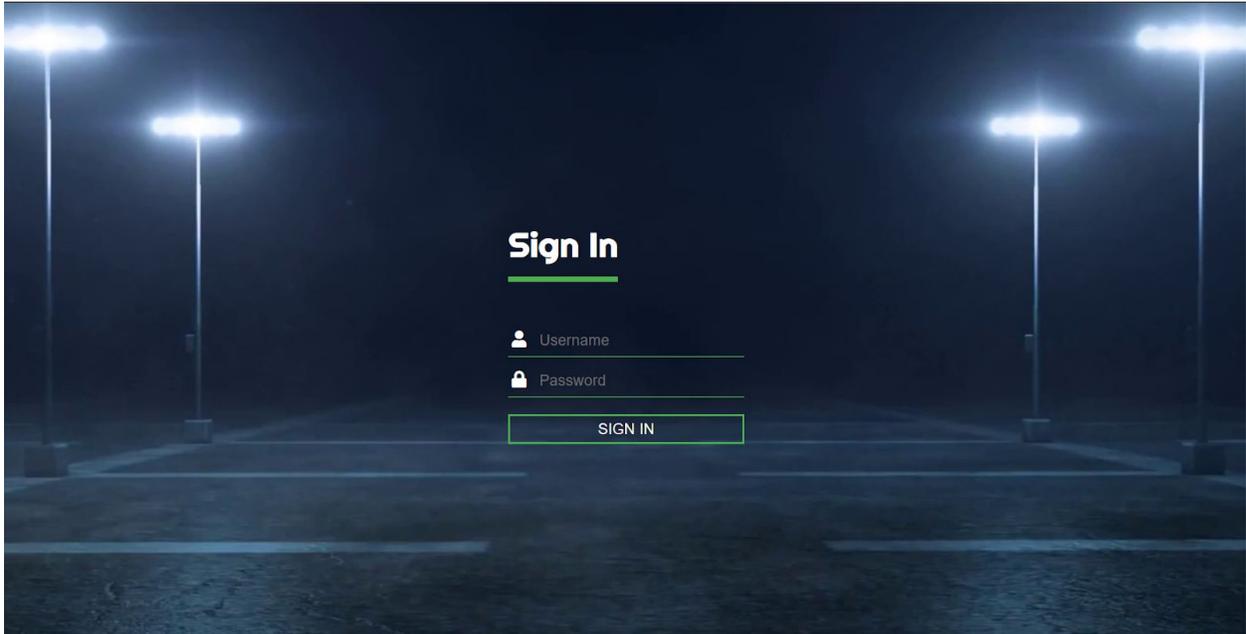
A. Preliminary Design:

3.2.1 Initial Design:



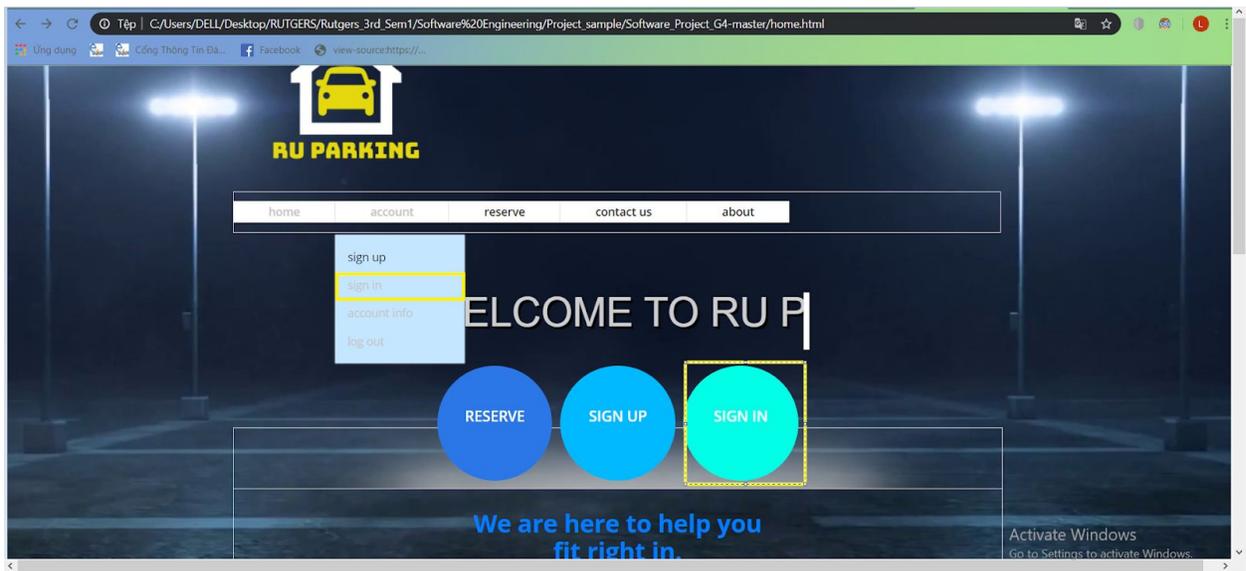
The image shows a preliminary design of a sign-in form. The form is centered on a blue background. It consists of a yellow header box with the text "Sign in". Below the header are two white input boxes with black borders, labeled "Email" and "Password". Underneath these is a checkbox labeled "Keep me logged in". At the bottom is a yellow "Submit" button.

3.2.2 Basic Programming Design:



A. Effort Estimation:

Navigation: total 1 click on Sign In either on the circle or on the navigation bar:



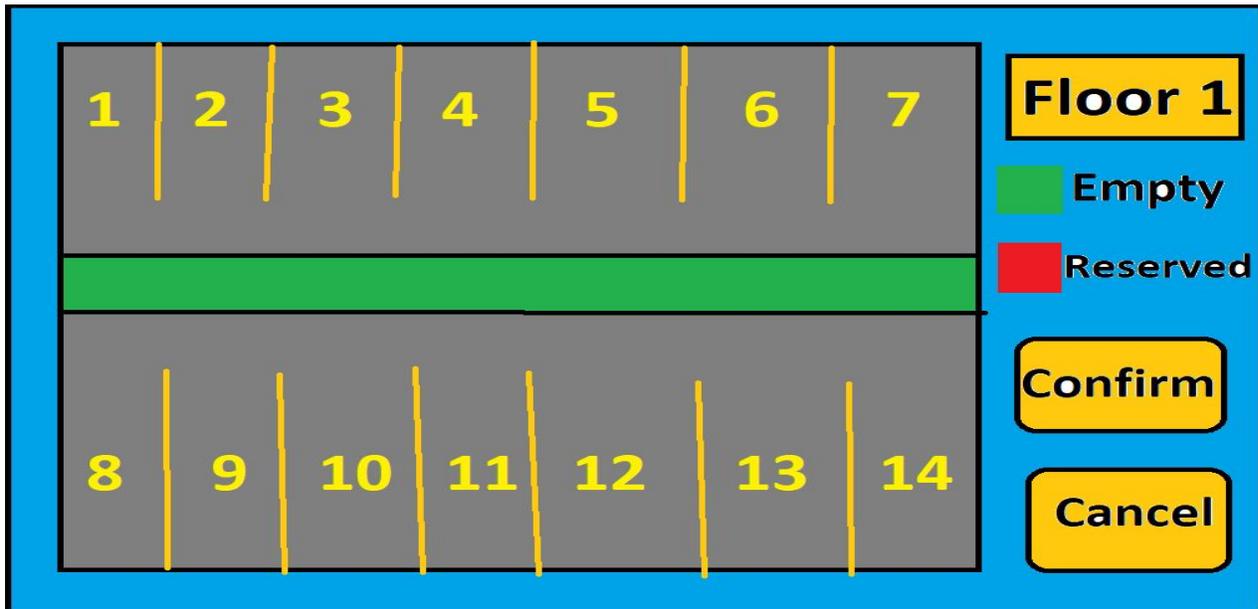
Data Entry: Total 1 click and 2 keystrokes as follows:

1. Click Sign In either on the circle or navigation bar then enter information:
 - Email (This is Username)
 - Password
2. Click button “Sign In”

3.3 Reservation - UC3:

A. Preliminary Design:

3.3.1 Initial Design:



A. Effort Estimation:

Navigation: total 3 step, undetermined keystrokes as follows:

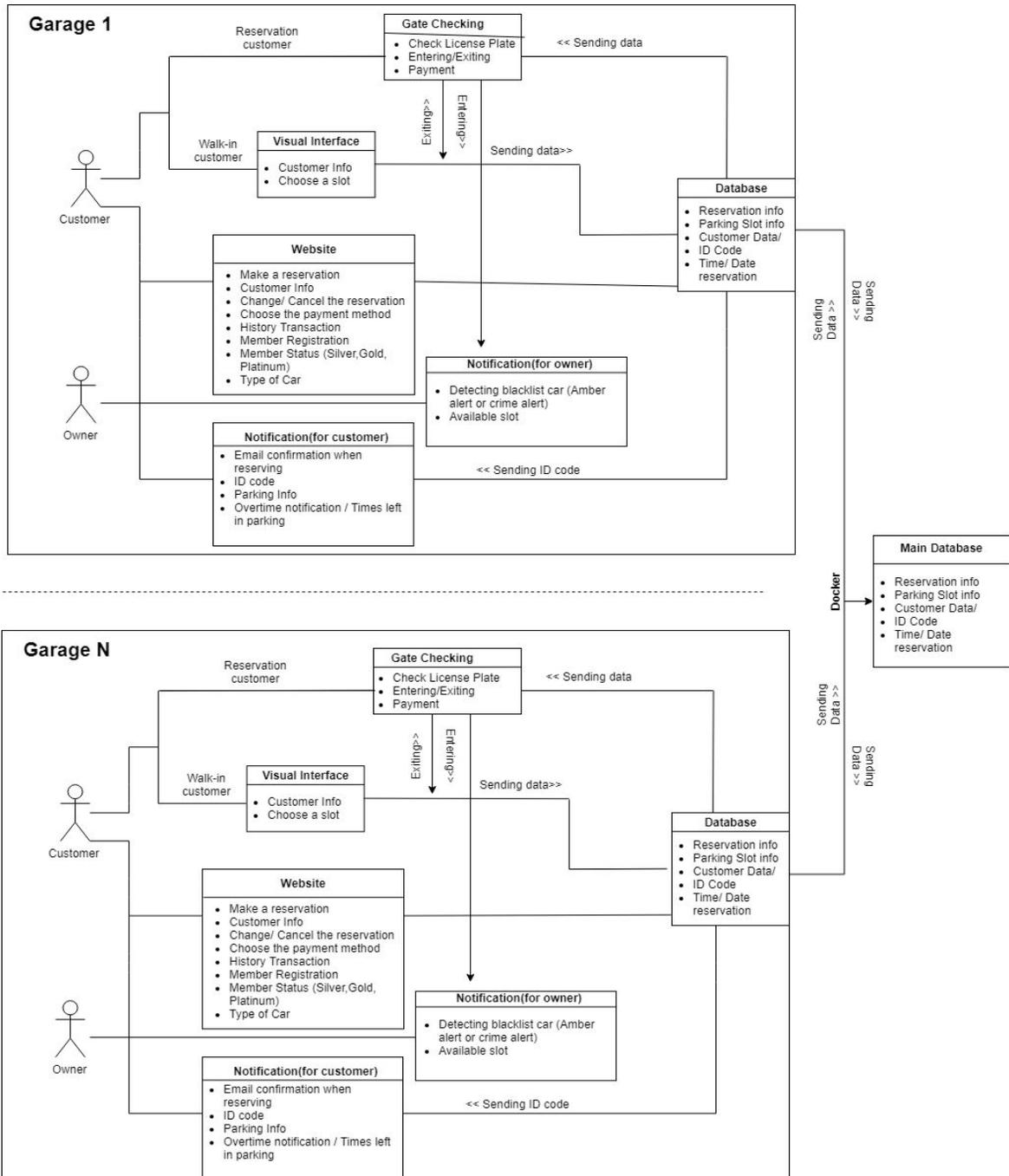
- Sign Up, create account
- Sign In to account
- Click Reserve

Data Entry: total is 3 keystrokes:

- Choose the level (level 1, level 2, level 3) (level is floor)
- Choose the lot
- Choose Date and Time then click Confirm

4. Domain Model Analysis

4.1 Domain Model



4.2 Domain Model Derivation

The following domain model is derived from the use cases and functional requirements with the highest priorities. The system with each use case allows the customer to register on the website (UC-1) and log in when complete registration (UC-2), then reserving for the desire slot (UC-4); on the other hand, regarding the walk-in customer which have the same concept as the web-register customer but reserving offline (UC-4) . Additionally, after reserving, the customer will park their car with the designed ID code (UC-20), keep track of the remaining time of the parking and pay for their bill or extend their duration time if the parking slot is available for that time. (UC-8, UC-9, UC-11, UC-12, UC-14, UC-18, UC-22,U-23)

There are four concepts that play a main role in the system: the camera system which having the license plate reader function which could define whether that is the customer which had reserved or the crime license plate and having the size detected function ; the visual or user interface that allows the customer to reserve and entering/exiting when reserving is complete and helping the customer to get into the desired slot; the notification system which sends the warning when detecting the crime license plate(for the manager) or sending the duration of time,the ID code,the reservation (for the customer); the pricing based on the user demands or the combined price when parking at multiple places in continuous time.

The four main procedures of the system:

- **The camera system:** When the customer enters the garage, at first sight the camera with the license plate reader will read the license plate and compare with the license plate in the database to check if the customer had reserved on the internet or not in terms of web-register customer and apply both when checking for the exit.
- **The visual and user interface:** When registering on the internet or reserving with the visual interface, the customer's information will be stored in the database and retrieving the data when implementing the license plate reader function. Moreover, the customer can register, log in and see the available slot at a given time to make,edit or cancel the reservations ; these actions will be saved in the system or in other words, the database.
- **The notification system:** When the registration is complete, the system will send an ID code, the slot and the hour parking for the customer to park in the right place and when entering the slot, the customer will need to show the ID code to acquire the lot. Moreover, during the parking time, the customer will receive the email notify the remaining time to make sure that the customer can take the car in time. And for the manager, sending a warning to notify if the blacklist car is entering the parking lot in

order to increase the safety for the customer and for the owner. Moreover, to notify the owner if a slot is available when the customer leaves the parking lot.

- **The pricing system:** When completing the registration, a price will be shown in the user interface and the price will depend on if the hour is rush hour or not, the adequate slot, the shape of the slot, the customer can choose to pay there if they use the system once. However, when the customer wants to park in continuous time, the customer can reserve through user interface or visual interface and complete the payment which will be the total price of each parking hour when leaving.

4.2.1 Concept Definitions (D-doing; K-knowing; N-neither)

| Responsibility Description | Type | Concept Name |
|---|------|-----------------------------|
| To check if the incoming customer has a reservation. | K | Gate Checking |
| To check if the incoming vehicle has a black-list plate. | K | Gate Checking |
| To estimate the size of the incoming vehicles. | D | Size Detecting Camera |
| To obtain the customer information and make sure the online reservations. | N | Visual/User Interface |
| To check if the exiting vehicles have finished the payment. | K | Gate Checking |
| To change or cancel the reservation before the desired time. | D | Website |
| To choose the payment methods by customers | K | Website |
| To manage customers and employee information and status. | K | Website |
| To record the history transaction. | N | Website |
| To send an email confirmation when reserving. | D | Notification (for customer) |

| | | |
|---|---|-----------------------------|
| To send an ID code for each parking. | D | Notification (for customer) |
| To notify customers if the parking is overtime. | D | Notification (for customer) |
| To show time remaining and real-time price to customers. | N | Notification (for customer) |
| To notify owners if there is any black-list car detected. | D | Notification (for owner) |
| To notify the number of available slots. | D | Notification |
| To obtain reservation information. | K | Database |
| To obtain parking slots information. | K | Database |
| To obtain ID code for incoming or reserving customers. | K | Database |

4.2.2 Association Definitions

| Concept Pair | Association Description | Association Name |
|--------------------------------|--|--------------------|
| Gate Checking↔Database | When vehicles entering, the gate checking checks the reservation information, black-list related information and the estimated size of the incoming vehicles, then sends the data to the database. When vehicles exiting, the database sends the vehicles' remaining time, the parking charge and the payment method to the gate checking. | Sending data |
| Gate Checking↔Visual Interface | The gate checking would check if the customers is entering or exiting the parking gate, if entering, the customers need to choose a slot from the visual interface. | Entering & Exiting |
| Visual | The visual interface send the information of the | Sending data |

| | | |
|---------------------------------------|--|------------------------------------|
| Interface↔Database | customers combined the chosen slots to the database. | |
| Website↔Database | The customers could make a reservation or change/cancel the reservation on the website. They can also choose their payment method on the website as well. When the reservation status is changed, the information will be sent to the database and the data there will be updated. Also, the database sends the history transaction data to the website. | Reservation/Payment status changes |
| Gate Checking↔Notification(for owner) | When doing the gate checking, if a black-list car is checked, then the owner will get a notification from the gate checking. The gate checking will also notify the owner if there is no parking slot left. | Alarm notification |
| Notification (for customer)↔Database | After reserving, the customers will receive an email confirmation and an ID code from the database. In the meanwhile, if the customer is parking overtime, the database will generate an overtime notification to remind the customer. | Sending ID code & overtime alert |
| Database↔Main Database | All the data in the every-garage database has an interaction with the main database. | Sending data |

4.2.3 Attribute Definitions

| Concept | Attribute | Attribute Description |
|------------------|---------------------|--|
| Gate Checking | Check License Plate | Check if the incoming vehicle is reserved and black-listed. |
| | Entering/Exiting | Check if the vehicle is entering or exiting the gate. |
| | Payment | Check the parking charge to be paid and also the payment method. |
| Visual Interface | Customer Info | Gather the customers' information. |

| | | |
|------------------------------|--------------------------|--|
| | Choose a slot | Choose an available slot for the customer on the basis of the size of the vehicle. |
| Website | Make a reservation | Make a reservation on the website. |
| | Customer Info | Gather the customers' information. |
| | Change/Cancel | Change/Cancel the reservation. |
| | Payment Method | Choose/Change the payment method. |
| | History Transaction | The history of transaction could be checked on the website. |
| | Member Registration | The customer could register a member card on the website for the parking discount,etc. |
| | Member Status | The members can check their member status on the website and check the members' privilege. |
| | Type of car | The information of the type of the car can be uploaded on the website for a better reservation. |
| Notification (for owners) | Detecting black-lisk car | The owner will get a notification if there is a black-list car detected. (from amber alert or crime alert,etc.) |
| | Available slot | The owner will receive a notification if there are a few parking slots left. The owner can also change their charging standard based on the number of available slots. |
| Notification (for customers) | Email confirmation | The customers will get a confirmation email after the reservation to make sure the security. |

| | | |
|-----------------|-----------------------|--|
| | ID code | Every vehicle has a unique ID code for corresponding its own parking information. |
| | Parking Info | The parking information like the place of the selected slot, the parking charge, etc. |
| | Overtime/Time left | The customers will receive a notification if the parking time is over the desired time. They may also be notified if there is, for example, five minutes left for their parking. |
| (Main) Database | Reservation Info | Store the information that if the customers have already a reservation and the desired parking time and parking garage. |
| | Parking slot Info | Store the information that if a specific slot is used or not. Also store the information of how many slots are being used at that time. |
| | Customer Data | Store the customers' information. |
| | ID Code | Every vehicle has a unique ID code for corresponding its own parking information. Store and generate the ID code. |
| | Time/Date reservation | Store the exact time and date of the reservation for customers. |

4.3 Traceability Matrix

| PW | Use cases | License Plate Reader | Visual interface | Website | Database | Notifications (Manager) | Notifications (Customer) | Gate Checking |
|----|-----------|----------------------|------------------|---------|----------|-------------------------|--------------------------|---------------|
| 3 | UC-1 | | | X | | | | |
| 3 | UC-2 | | | X | | | | |
| 3 | UC-3 | | | X | | X | | |
| 4 | UC-4 | | | X | | | | |
| 4 | UC-5 | | | X | | | | |
| 8 | UC-6 | | X | | | | | |
| 4 | UC-7 | | | X | | | | |
| 7 | UC-8 | | | | | | | X |
| 5 | UC-9 | | | | | | | X |
| 6 | UC-10 | | | | | | | X |
| 4 | UC-11 | | | | | | X | |
| 4 | UC-12 | | | | | | X | |
| 3 | UC-13 | | | | | | | |
| 4 | UC-14 | | | X | | | | |
| 3 | UC-15 | | | | | | | X |
| 3 | UC-16 | | | | | | | X |
| 8 | UC-17 | X | | | | X | | X |
| 7 | UC-18 | | X | X | | | | |
| 8 | UC-19 | | | | | | | |
| 6 | UC-20 | | | | | | | |
| 8 | UC-21 | | | | | | X | |
| 7 | UC-22 | | | | | | X | |
| 5 | UC-23 | | X | X | | | | |

4.4 Mathematical Modeling

An average transfer of data between any two actors has a man-in-the-middle risk, where a third bad actor can tap the message midway and see what is inside. To make such attempts at theft meaningless, we will use asymmetric encryption to secure various forms of transactions between us and our customers; this will protect key customer information by making any records unreadable and virtually uncrackable.

We will accomplish this by using the following mathematical technique:

1. We will give the customer a *public key*, which is the left side of the equivalency:
 $m^E \text{ mod } N \equiv c$
2. The customer will replace “m” with their intended message using a padding scheme and send us back the encrypted message “c”.
3. We will decrypt the message using our *private key* and use the content of the message. The message could be the license plate number of the customer’s car, for example.
4. Following the example in step 3, we would then record this transaction in our entry/exit ledger (in its encrypted form).

5. In the event of the theft of this ledger, the customer's privacy would be safe and their travel history will not be exposed. This is due to the fact that the bad actor does not have the *private key* required to make actual meaning of all the data in the ledger.
6. Any bad actor who decides to brute-force decrypt the data would be stuck solving the puzzle for decades or longer.

The determination of the perfect private key is where the largest amount of mathematical modeling is required. We have to make sure that even if the public key is known, the private key cannot be cracked using it. We also have to make sure that the private key cannot be easily cracked. In order to achieve these objectives, we would have to pick the right values for “N”, “E” and a third letter “D”, which will be used to undo the effects of “E”.

“E” can be any small number greater than 1 of our choosing, with the only condition being that it cannot share any factors with $\phi(N)$, which is something we'll explore soon.

To find “N”, we will pick any two really large, preferably 100 digits or more, prime numbers and multiply them together:

$$N = P_1 * P_2$$

To understand why this is the best idea, we will first have to define something called the breakability of a number “X”. The breakability of a number is the amount of numbers $\leq X$, but greater than 1, that do not share any common factor with X.

The best “X” to pick therefore is a prime number. This is because a prime number cannot be broken up more than twice, once with a factor of 1 and another with a factor of itself. This property makes it so that there will be no ambiguities in future calculations.

Here is the formal definition, when X is a prime number:

$$\phi(X) = X - 1$$

The following property is also valid for the multiplication of two prime numbers :

$$\phi(A * B) = \phi(A) * \phi(B)$$

We can then port this information with “N” this way:

$$\phi(A * B) = \phi(A) * \phi(B)$$

$$\phi(P_1 * P_2) = \phi(P_1) * \phi(P_2)$$

$$\phi(N) = (P_1 - 1) * (P_2 - 1)$$

This part is extremely difficult to do in the other direction and is the reason, why we would want to keep it $\phi(N)$ secret.

The number “D” is another number we would want to keep secret. Why? Because as we will see, this number is really the one that will unlock the encrypted message easily and in one calculation. This is our *private key code*.

To begin to find “D” we must define Euler’s Theorem. This theorem utilizes $\phi(N)$, the breakability of N, that we found from a previous part in its definition.

Here is how the Theorem is defined, using our variables:

$$m^{\phi(N)} \equiv 1 \pmod{N}$$

Using this and some properties of modulus mathematics, we can logically deduce the following:

$$m^{k*\phi(N)} \equiv 1 \pmod{N} ; \text{ because } 1^k \text{ always equals } 1, \text{ not matter the } k.$$

$$m * m^{k*\phi(N)} \equiv m \pmod{N} ; \text{ because any number times } 1 \text{ equals itself.}$$

$$m^{k*\phi(N)+1} \equiv m \pmod{N} ; \text{ using basic rules of exponents}$$

We can use this definition on the original public key: $m^E \pmod{N} \equiv c$, and find the following, assuming that we’re looking for a “D” that depends on $\phi(N)$:

$$m^{D*E} = m \pmod{N}$$

$$\text{Then, } D = (k * \phi(N) + 1)/E$$

All this means is that our secret $\phi(N)$ factorization has been utilized in a way that incorporates the non-secret parts E and N. We can now use D to transform the encrypted “c” message to the decrypted message “m”.

In conclusion, by incorporating all the steps, we will have an extremely strong encryption system. This will put the customer’s mind at ease, knowing that even if ledger data is leaked, their information is safe.

5. Project size estimation

| User Case | UC Name | UC Description | Estimated Size |
|-----------|--------------------------|--|----------------|
| UC1 | Sign up | Customers create an account on the website. | 4 |
| UC2 | Log in | Customer use their registered credential to access the website. | 3 |
| UC3 | Slot Availability | Check if slots are available for parking. | 5 |
| UC4 | Slot Preference | Select from the available parking slots as per choice. | 5 |
| UC8 | Dynamic Pricing | Charging customer depending on the time of the day when the demand if high/low. | 6 |
| UC9 | Payment | Charge your customers at the end of parking tenure. | 5 |
| UC11 | Time elapse | To keep an account of time elapsed since parking. | 5 |
| UC12 | Time update | Send customer update about the time left before reservation expiry. | 7 |
| UC15 | Arrival | Customer entering the garage. | 4 |
| UC16 | Departure | Customer exit the garage. | 4 |
| UC17 | Blacklisted Car Alert | Inform garage manager about blacklisted car detection. | 5 |
| UC18 | Suitable Garage Location | Choose the preferable parking location. | 7 |
| UC20 | Parking Security | Send code to the customer when making online reservations or offline, when the customer gets to the desire-slot, to get into the slots, the customer needs to give the code to the system and then the system will allow the customer to get in. | 8 |
| UC22 | Email Updates | Send pay amount, security code, time elapsed and start of buffer time. | 6 |

6. Plan of Work

Below is the gantt chart diagram which enlists detailed task breakdown and the corresponding estimated start date and end date. It should be noted that demo 1 and demo 2 has task duration of a day and hence has no visual bar.

Description of each task is also given below.

6.1 Gantt Chart

| | Start Date | End Date | Timeline | Status |
|--------------------------------------|------------|------------|----------|----------|
| Automated Garage Parking | 09-20-2019 | 12-12-2019 | | |
| Design Discussion | 09-20-2019 | 9-25-2019 | ■ | Complete |
| Initial Design | 09-25-2019 | 9-28-2019 | ■ | Complete |
| Design Review | 09-28-2019 | 10-2-2019 | ■ | Complete |
| Detailed Design | 10-03-2019 | 10-6-2019 | ■ | Complete |
| Analysis of implementable Tools | 09-28-2019 | 10-4-2019 | ■ | Complete |
| Develop System Modules (1st Phase) | 10-04-2019 | 10-22-2019 | ■ | Active |
| Integrate System Modules (1st Phase) | 10-22-2019 | 10-25-2019 | ■ | Upcoming |
| Perform System Testing | 10-25-2019 | 10-27-2019 | ■ | Upcoming |
| Document Bugs Found | 10-25-2019 | 10-27-2019 | ■ | Upcoming |
| Debug | 10-25-2019 | 10-29-2019 | ■ | Upcoming |
| First Demo | 10-30-2019 | 10-30-2019 | | Upcoming |
| Review Feedback | 11-01-2019 | 11-4-2019 | ■ | Upcoming |
| Redesign | 11-04-2019 | 11-8-2019 | ■ | Upcoming |
| Develop & Integrate (2nd Phase) | 11-08-2019 | 11-26-2019 | ■ | Upcoming |
| System Testing | 11-27-2019 | 12-3-2019 | ■ | Upcoming |
| Document Bugs Found | 11-27-2019 | 12-2-2019 | ■ | Upcoming |
| Debug | 11-27-2019 | 12-4-2019 | ■ | Upcoming |
| Deployment | 12-4-2019 | 12-10-2019 | ■ | Upcoming |
| Second Demo | 12-12-2019 | 12-12-2019 | | Upcoming |

6.2 Task Phase Description

Design Discussion and Initial Design

- Basic UI design for sign up, sign in, registration and reservation page.
- Establish routes for each user case (determine which page goes to where)
- Preliminary ideas for block chain assisted security of user information
- Determine the containerization technique for information exchanges

Design Review and Detailed Design

- Finalize the UI (HTML & CSS)
- For each use case determine user page interaction flow
- Research and simulation of the analytical model for block chain assisted information privacy
- Black listed car detection
- Determine detailed module interaction and object finalization of each module

Analysis of Implementable Tools

- Determine the tools that would be used in container deployment (docker, lxd, kubernetes) and thoroughly understand the features of each of these platform-as-a-service product.

Develop and Integrate System Modules

- REQ1 and REQ2
- Car detection at the entrance and customer entry after reservation confirmation
- Depending on the spot choice bring the customer to the correct level
- Verify if the vehicle is parked in the reserved slot
- Offer slot choices to walk-in customers
- Email notification with reservation ID to the customer
- Charge customers depending upon dynamic pricing
- Encrypt customer payment and information details
- Depending upon customers parking history for the day provision combined pricing
- Provide information to the customer about parking slot availability at other garage location

System Testing

- Code is implemented completely and checked against our requirements whether it is addressing our needs which are gathered in the design and analysis phase or not.
- All the bugs of our software are tracked and fixed until we obtain our desired product with the defined quality standards.

Deployment

- Our product will be deployed for beta testing for finding bugs
- Depending on the users feedback the our system can be improved

6.3 Product Ownership

| Member | Done | In Progress | Future Responsibility |
|--------------------------|--|---|---|
| Shalini & Khanh | Initial container configuration Creating containers Pushing files into the container Car number plate detection | Number plate recognition from video streaming | Running the automatic number plate detection app in the container Storing pricing and slots availability information in the container Migrating the container from one node to another (for information availability at other garage locations) |
| Luan, Duc, Tan, Zhuoyang | Creating website contains the main page with sign up page, sign in page. | Coding the map for parking lot and create reserve page (reserve form), mark the lots that have been reserved. Adding well designed web page layout for user friendly experience | Using Django (Python) to deal with back-end. Process database. Create confirmation page and successful reservation notification, log out features. Create pages for admin. Integrating all the modules |
| Nainil, Tan | Creating the barebones of the post-sign page, where payments and reservations will be made. | Building an encryption system to protect customer data. Building a Database to store key information for all parts of the project. | Building a fully-functioning blockchain to update and maintain ledgers of all transactions. Connecting the database with the finished website and plate recognition systems. |

7. References:

Books:

- https://www.ece.rutgers.edu/~marsic/books/SE/book-SE_marsic.pdf

Reports:

- [Group#3_2018_project](#)
- [Group#5_2013_project](#)

Web Resources:

- <https://www.devteam.space/blog/how-to-integrate-a-blockchain-technology-into-your-project/>
- <https://www.redhat.com/en/blog/container-migration-around-world>
- <https://www.netapp.com/us/info/what-are-containers.aspx>
- <https://lxd.readthedocs.io/en/latest/>
- <https://ssd.eff.org/en/module/deep-dive-end-end-encryption-how-do-public-key-encryption-systems-work>
- <https://fortune.com/2018/06/27/facebook-data-privacy-blockchain/>

UML Design:

- <https://www.draw.io/>