Parking Garage Automation

Corey Chen
Chunhua Deng
Jonathan Garner
Siyu Liao
Siddharth Musale
Ridhima Sakhuja
Xianglong Feng

Project blog: https://sites.google.com/a/scarletmail.rutgers.edu/parking-garage-automation/

Table of Contents

1. Statement of Work & Requirements

1.1 Problem Statement

The majority of modern parking garages have inherent flaws that cause them to be less efficient than what they could be. A recent study indicates that drivers spend an average of 17 hours per year looking for parking and approximately $97 per driver[1]. This situation is only exacerbated during peak hours, when a high influx of cars causes congestion in parking garages, resulting in more time and money wasted on looking for parking. Many drivers are also unsatisfied with the flat rate pricing system, where they are charged a fixed hourly rate regardless of the time of day. Additionally, a majority of parking garages do not have a system that lets their customer keeps track of how long their car has been parked for and how much they will have to pay when they leave. Along with this, parking garages also lack a security system that helps ensure the safety of the customers' cars. In metropolitan cities, parking garages are often located in unsafe areas, where anyone can enter and leave the garage regardless of whether they are a paying customer of not. This smart-garage proposal contains ideas to improve customer experience of the parking garage, save wasted time in the garage by looking for empty spots, and boost revenue for the garage.

1.2 Proposed Solution

In order to provide a more streamlined process for our customers, our plan is to implement an automated system that will allow customers to view and reserve available spots using a visual interface hosted on a website. This will give customers a simplified view of available parking spaces that is easily accessible on both computers and mobile devices, and allow them to reserve or change spots at any time. Customers will be able to reserve spots for a specified check in time and reservations for an occupied spot will open once the parked patron leaves the spot. There will be long term term parking as well as short term reservations which will allow customers with a variety of needs to utilize the parking garage.

Our proposed system includes multiple novel ideas that have not yet been implemented in previous automated parking garage projects. The first of these is a visual display on the website that will allow customers to view and select spots on a simplified map in order to check which spots are currently open or to place reservations. This feature will primarily be available to customers who come in off the street for ad-hoc parking through the use of a screen at the check-in area of the garage that will display available spots and allow customers to select the spot they want. Another new idea is our smart pricing system, which will continuously update

the cost of spots based on multiple factors (time of day, amount of reservations, etc). Along with these, we are proposing a traffic monitoring system that keeps track of how many cars are driving to their spots on each level of the garage and limits the number of active cars per floor in order to cut down on traffic within the garage. Another newly proposed feature is an improved security system that will limit garage access to parked customers only in an effort to increase the feeling of safety within our customers. A final feature we are proposing is the use of a unique alphanumeric security codes that customers receive either via email or SMS in order to enter or exit the garage. This guarantees that only customers with security codes can re-enter the building, protecting the customers' vehicles from non-customers. All the features mentioned above are further elaborated on down below.

Due to the fact that ad-hoc parking is a major component of a parking garage's income, there will be a limited number of reservable spots. These spots will be separated by floor or section in the garage. This means that even if reservations are fully booked, patrons will be able to still drive up and park if there are ad-hoc spaces available. The price of parking in these spots will be determined by the smart pricing system (explained more below) which will generate more revenue.

Another major component of streamlining the parking garage is reducing traffic inside and allowing customers to easily find their spot and park. When a customer enters the garage for ad-hoc parking, they will be shown the same display that is available on the website via a monitor and prompted to choose an available spot using the visual interface. This system will display the relative location of the selected spot on the visual interface, which will cut down on the time it takes for drivers to locate their spot. To further cut down on traffic in the garage, drivers will have to get to their assigned floor using a 1-car sized elevator. This elevator will scan the license plates of each car that uses it and communicate with a system that will keep track of which cars are currently on which floors, and whether or not they are parked. Each spot also contains a sensor which can tell whether or not a car is occupying the spot, and report that information back in order to tell whether or not the car that is active on that floor has parked yet. As soon as it is determined that the car on a specific floor has parked, the next car will be allowed in to locate their spot and park. This system aims to improve a customer's ability to find their spot and park in it.

This proposal also includes a smart-pricing system that entices more customers to use the garage. This pricing system we propose includes incentives for customers to register for membership of the garage by offering special deals such as reduced fees for longer term members. This pricing system dynamically adjusts the hourly rate of parking so that the price of parking spots goes up as the demand for them grows and diminishes as the parking garage is utilized less. This is intended to help make more profit during rush hours as well as maximizing customers during the other times of the day. However, we also offer the fixed rate strategy for someone preferring traditional parking billings. We can give fee estimations and save money for customers by helping them choosing fixed rate or dynamic rate.

Our proposed system will also consist of a notification system that will regularly update the customer regarding their status via text message or email. Upon arrival, once the driver has parked their car at the designated parking lot, this system will send a message confirming that they have parked in the correct location and a ticket that will have a confirmation code that will be required to leave and enter the parking garage. The ticket will also include information such as the driver's arrival time and the parking lot number. After every hour, the notification system will send a text message to the customer, letting them know how long it has been since they have left their car at the parking garage. Not only will this allow customers to keep track of their parking spot, but it also be useful in trying to ensure that drivers with reservations do not overstay their allotted time.

Another concern of many customers is the security of their car when parked inside a garage. In an effort to cut down on potential crime or theft in the garage, a system will be put into place that will only allow verified customers in and out of the garage on foot. When a customer parks their car, they will be sent a notification through the previously mentioned system containing a unique confirmation code. In order to unlock the pedestrian doors and enter or exit the garage, users will be required to enter their access code. This system aims to enforce a feeling of safety and security in consumers.

The main goal of this proposed project is to optimize our current parking garage system and turn it into an automated, web-based system. Doing so will allow us to cut down on many operating costs, such as payroll for check-in area attendants who will no longer be of use. This system will also allow us to view and change current reservations, as well as adjust the pricing of spots manually in order to capitalize on rush hours and bring in more revenue. However, these features will strictly be available to administrative users. The typical customers who will be parking in the automated garage will fall under two general categories: those who plan to park long term (consumers who may be on vacations, business meetings, etc.) and those who plan to park short term (visitors in the city, commuters, etc). The customers we are targeting are those who are willing to possibly pay a higher price for parking for the added convenience of being able to view and reserve spots online. Many commuters in busy cities would be willing to pay higher premiums for the security, efficiency, and convenience our automation provides.

In terms of metrics that will measure the success of the automated garages, we will be looking mostly at revenue brought in by the smart pricing system and the average number of spots being used. In addition to these quantitative metrics, we will also track the number of users who register for membership on the website and how many users use the website reservation system in order to show whether or not customers are taking advantage of the available features.

There are three overarching goals that this system is trying to achieve. Firstly, there will be a website that interacts with the database in order to provide services to members and non-members. This will include reservation services and deals for members of the garage as well as a user interface where ad-hoc customers can select preferable spots to park. In addition, this system will attempt to maximize profits through a dynamic, smart-pricing algorithm based on the

occupancy percentage of the garage and time of day. Lastly, the system will also introduce another algorithm that minimizes traffic blockage through the whole garage by bottlenecking the number of cars that are searching for parking spaces on each floor.

Functional Features (Summary) - Users will be able to:

1. View and reserve spots online
2. Select spots for ad-hoc parking
3. Locate their spots using a visual interface
4. Cancel reservations or switch spots at will
5. Receive notifications such as warnings based on time spent parking or confirmation of parking in the correct spot
6. Enter or exit the building using a unique confirmation code

1.3 Novelty

We investigated the report of previous works including the one in Spring 2012 and the one in Spring 2013. To have a clear overview of those works and to distinguish ours contribution from existing works, we will give a summarization of those works and compare with them. Following are features listed in the report of Spring 2012:

1. Place online reservation in the form of a website
2. Website Registration
3. Security guard helps customer pay bills and exit the garage
4. Notify user that they are not identified
5. Notify operator if sensors not working
6. Park with rented or borrowed vehicles
7. Garage remodeled such that an elevator can lift vehicles to different decks
8. One-way entering and existing system to avoid congestions
9. camera based license plate recognition software to track vehicles as they enter and exit the garage
10. the license plate must be recognizable for elevator to work
11. Allow walk-in customers
12. Support confirmed reservation (no credit card) and guaranteed reservation (with credit card)
13. 30-minute grace period for customers not showing up, after which spots will be unreserved
14. Floor sensors for detecting occupancy
15. Elevator will lift up vehicles to the correct spots

Besides, following are the features listed in the report of 2013.

1. It has an online reservation system.
2. It implements a license plate reader reader.
3. It allows customers to create accounts to store personal information for fast parking
4. The system has three types of account, which is customer account, manager account, and employee account
5. It can notify potential customers if there are available parkings spots and for how long they can park
6. It broughts cars to a checkpoint and take pictures of the car before it enters the garage and right as its about to leave for security reason.
7. It can automatically detect frequent customers by reading license plates.
8. It integrates a simulator to test the system
9. It deals with overstate problem.
10. It encrypts its user data
11. It backs up its account information, parking data and daily reports once a day

When compared with existing works, the main novelty of this project are listed as follows:

1. Smart Pricing System: We charge users accordingly in order to attract users and provide for their convenience. For example, we charge higher when less available spots and charge lower when more available sports.
2. Spot viewing interface: This interface is primarily meant for the ad-hoc customers of the garage. Instead of just assigning spots to a customer, customers will be able to view an interface and select spots based on their personal preference instead of getting forced into a spot they may or may not want.
3. Different Congestion Control: Our system will work in a way such that there will only be one vehicle parking in to its spot concurrently on each floor. This reduces the amount of traffic on each floor and helps us to ensure that people occupy their specified parking spaces.
4. Automatic Security: After customers park in their specified spots, they will be sent an alphanumeric code via email or SMS that allows them to access the parking garage. The code can be used to both enter and exit the garage. This guarantees that only customers and people travelling with customers can go in and out of the garage, guaranteeing the security of their vehicles at least against non-customers.

1.4 Glossary of Terms

- Account - Holds all of a member's information, such as username and contact information
- Ad-Hoc Customers - Customers who drive in off the street for unplanned or unreserved parking
- Administrative User - A high-level user who will have permissions on the website to log-in and view or change management information such as reservations, pricing, etc.
- Check-In Area - The first area customers encounter upon entry into the garage with gates for entry
- Confirmation Code - Code sent to verified customers after parking to allow exit and reentry into the parking garage
- Customer - Person who enters the garage with the intent of parking in a spot (with or without reservation)
- Drivers - Customers in cars that are currently parking or driving in the parking garage
- Elevator - Platform used to raise cars to upper floors
- License Plate Scanner - Device which reads the license plate of a car waiting to enter the elevator
- Member - Customer who has registered on the website and is now a member with login information
- Notification - Alert or message sent through text or email to a customer
- Occupied Spot - Parking spot which currently has a car parked in it
- Parking Spot Number - Unique number which identifies a spot and its relative location
- Pedestrian Doors - Allow foot traffic in and out of the parking garage; can only be unlocked and used with a valid confirmation code
- Registration - The process by which a customer becomes a member
- Reservation - An agreement that the garage will hold a spot for a customer who intends to park there
- Reserved Spot - Spot which a customer has reserved with the intention of parking in it
- Sensor - Device in each parking spot that determines the presence of a car in that spot
- Smart Pricing System - Automatically determines the prices of reservations and ad-hoc parking spaces based on an algorithm and relevant information
- Ticket - Document containing a customer's parking and reservation information
- Traffic Control System - Keeps track of which floor has active drivers, and the spots in which these drivers belong in order to confirm that customers park in the correct location

- Username - Unique identifier created by a customer for themselves during the registration process
- Vacant Spot - A spot not currently occupied by a car and not currently reserved
- Verified Customer - Customer who has parked their car in the correct spot, as determined by the sensors and traffic control system
- Visual Interface - Graphical interface in the form of a simple map that displays the location of parking spots and whether or not they are occupied
- Website - Used by administrative users to view information and by customers to view spots and make reservations

2. Functional Requirement & User Interface

2.1 Stakeholders

This system is primarily intended for implementation in current garages to help garage owners increase efficiency and profits. This system will also be beneficial towards people who can utilize or maintain the system. Below is a list of possible stakeholders in the system.

- Parking Garage Owners
- Business Enterprises
- Customers: ad-hoc and walk-ins

2.2 Actors and Goals

Actors	Goals	Use Cases
Parking Interface	To display for customers on the website to allow them to reserve spots in the garage ahead of time	UC-2,UC-3,UC-4,UC-6
Parking interface	To display for ad-hoc customers to choose vacant spots to park in once they enter the garage	UC-2,UC-4,UC-6, UC-16
License-Plate Reader	To read the license plates of vehicles and passes it to the system	UC-9,UC-10
System	Update customer information and payment data	UC-2,UC-11,UC-14, UC-15

System	Update parking information in the garage	UC-2,UC-15,UC-16
Customer	To register in the system	UC-1
Customer	To review the available parking lots	UC-2
Customer	To enter the garage	UC-9
Customer	To leave the garage on foot	UC-8
Customer	To leave the garage by car	UC-10,UC-14
Customer	To make reservations on the website.	UC-1,UC-2,UC-3
Customer	To cancel his reservation	UC-7
Customer	To do Ad-hoc parking while choosing their spot	UC-2, UC-4
Customer	To update or modify account information	UC-11
Customer	To park in the garage	UC-5
Customer	To receive notifications	UC-15
Garage Owner	To choose a pricing strategy	UC-13
Customer	To pay bills for the parking service	UC-14
Garage Owner	To analyze statistics of the usage of the garage	UC-1,UC-3,UC-4, UC-13
Garage Owner	To keep track of parking spots in garage	UC-2,UC-15
Garage Owner	To manually input information in case of system failure	UC-13
Sensors	To monitor traffic on each floor	UC-5,UC-6
Sensors	To check to make sure customers have parked in the correct spot	UC-5,UC-6
Elevators	To bring customers to their designated floors	UC-5,UC-9
Elevators	To bring customers to the ground floor once they are leaving the garage	UC-9,UC-10
Security System	To ensure only customers and the people they are traveling with can enter or exit the garage	UC-8, UC-9,UC-10

Security System	To send customers an alphanumeric code they will need to enter the garage along with periodic updates about time elapsed	UC-10,UC-15

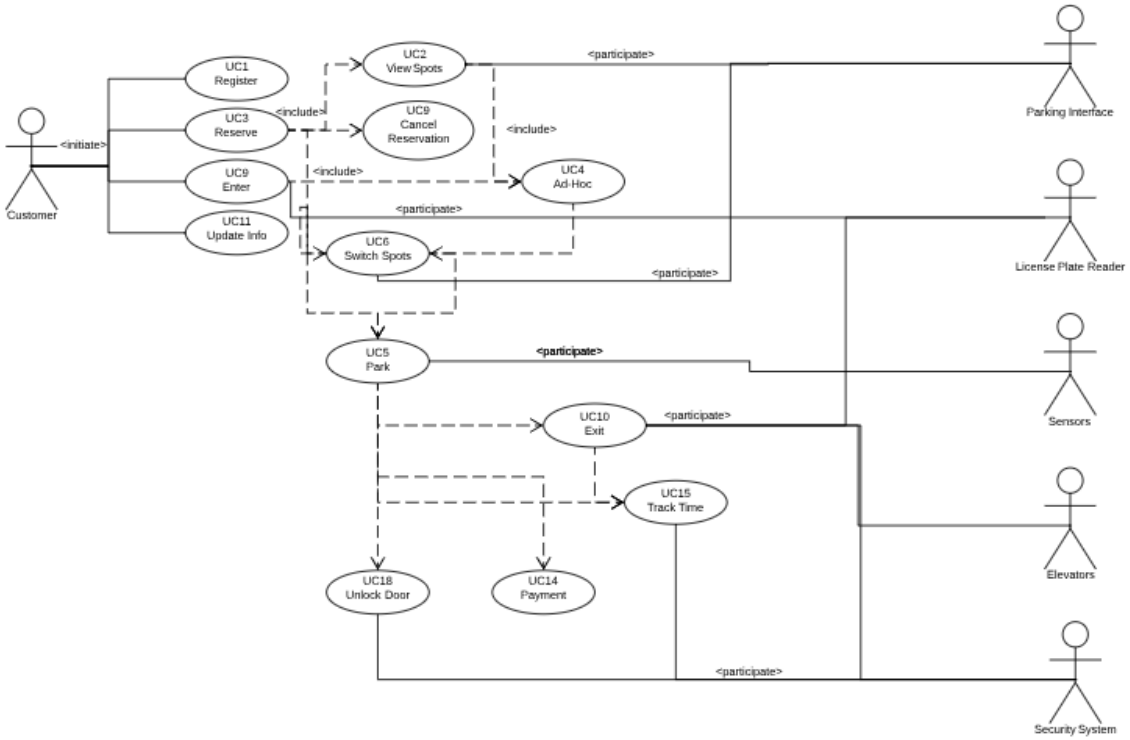
2.3 Use Cases

2.3.1 Casual Description

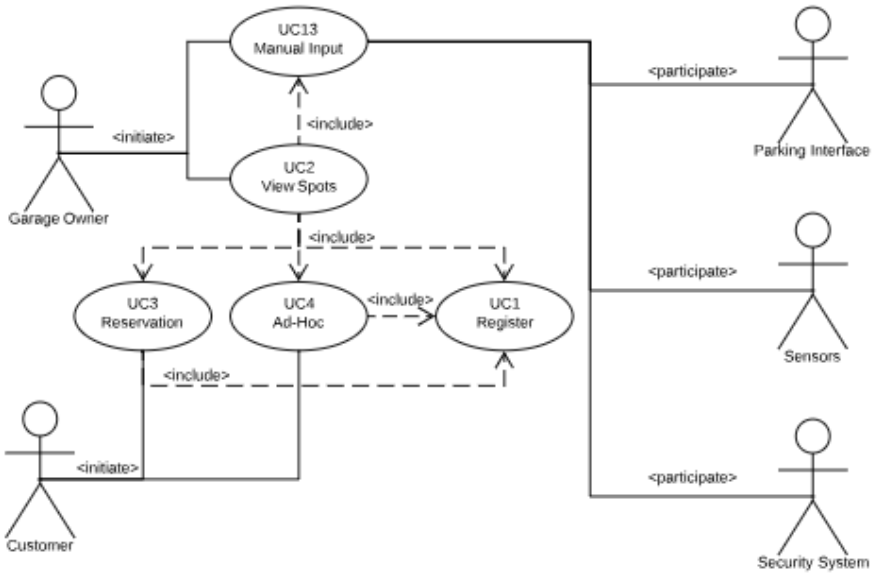
UC-1: Register- to register an account on the website
UC-2: View Spots - to view available and unavailable spots in a selected time window
UC-3: Reservation - to save a spot reservation online ahead of time
UC-4: Ad-Hoc - to select a spot at the garage with no reservation
UC-5: Park - to park a driver's car
UC-6: Switch spots - to change a reserved or selected spot before parking
UC-7: Cancel reservation - to cancel a reservation before parking
UC-8: Unlock door - to unlock and gain access to the pedestrian door to exit or enter the garage on foot
UC-9: Enter- for a driver to enter the garage in their car
UC-10: Exit - for a driver to exit the garage in their car
UC-11: Update information - to change account information on the website
UC-12: Set pricing - to override smart pricing options or set smart pricing options
UC-13: Manual Input - to input information directly in case of system failure
UC-14: Payment - to pay bill at the end of parking
UC-15: Track Time- to track how much time has elapsed since parking
UC-16: No Availability- to notify ad-hoc customers in case there is no availability in the garage

2.3.2 Use Case Diagram

Customer:



Owner:

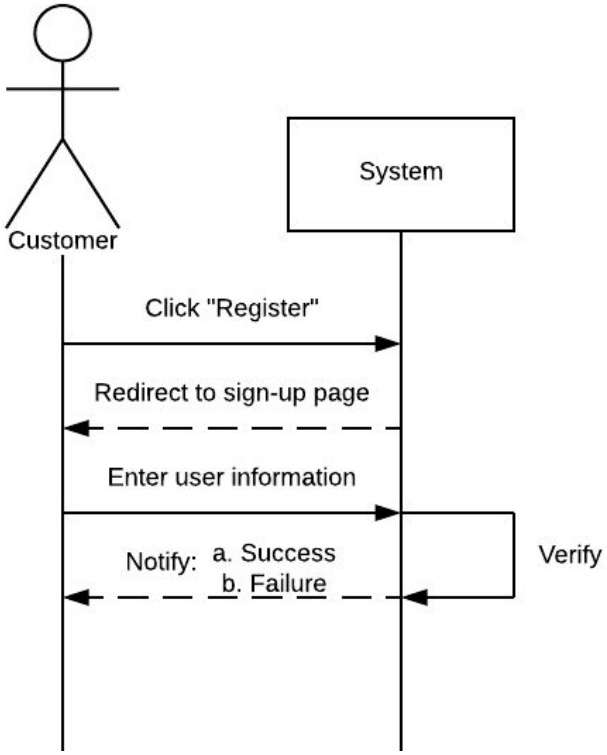


2.3.3 Traceability Matrix

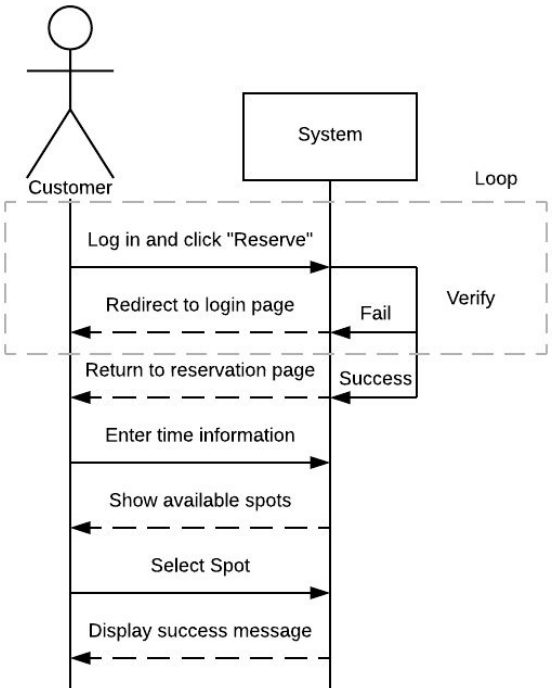
Req't	PW	UC1	UC2	UC3	UC4	UC5	UC6	UC7	UC8	UC9	UC10	UC11	UC12	UC13	UC14	UC15	UC16
REQ-1	3									X							
REQ-2	3			X	X					X				X			
REQ-3	5		X	X	X									X			X
REQ-4	5		X														X
REQ-5	4	X										X					
REQ-6	4	X		X								X				X	
REQ-7	5		X		X												
REQ-8	5												X		X		
REQ-9	4					X											
REQ-10	4					X											
REQ-11	3					X										X	
REQ-12	3					X										X	
REQ-13	4													X	X		
REQ-14	2												X				
REQ-15	4			X													
REQ-16	4								X	X	X						
REQ-17	3					X	X										
REQ-18	5			X													
REQ-19	2			X													
REQ-20	2							X									
REQ-21	4												X		X		
REQ-22	3							X									
MAX PW		3	4	4	3	5	2	2	1	4	5	3	4	2	5	4	2
TOTAL PW		20	15	23	25	17	3	5	10	10	4	15	20	10	23	10	10

2.3.4 Fully Dressed Description & Sequence Diagram

Use Case UC-1: Register
Related Requirements: REQ-05, REQ-18, REQ-24, REQ-25, REQ-27, REQ-32, REQ-38
Initiating Actor: Customer
Actor's Goal: To register an account on the website
Participating Actors: Customer, System
Preconditions: The system will request the necessary information from the customer on the registration page.
Postconditions: The customer's account will be stored within the database which will be backed up regularly
Flow of Events for Main Success Scenario: ->1. Customer accesses the website and chooses to "Register" an account <-2. The system returns a page that requests the necessary fields ->3. The customer enters information into the required data fields <-4. The system checks the username and email against the database. a. If the username or email is not unique, return to step 3 to alter login information. b. If the username and email are both unique, the system takes the information and enters it into the database. the system notifies the user for successful registration.

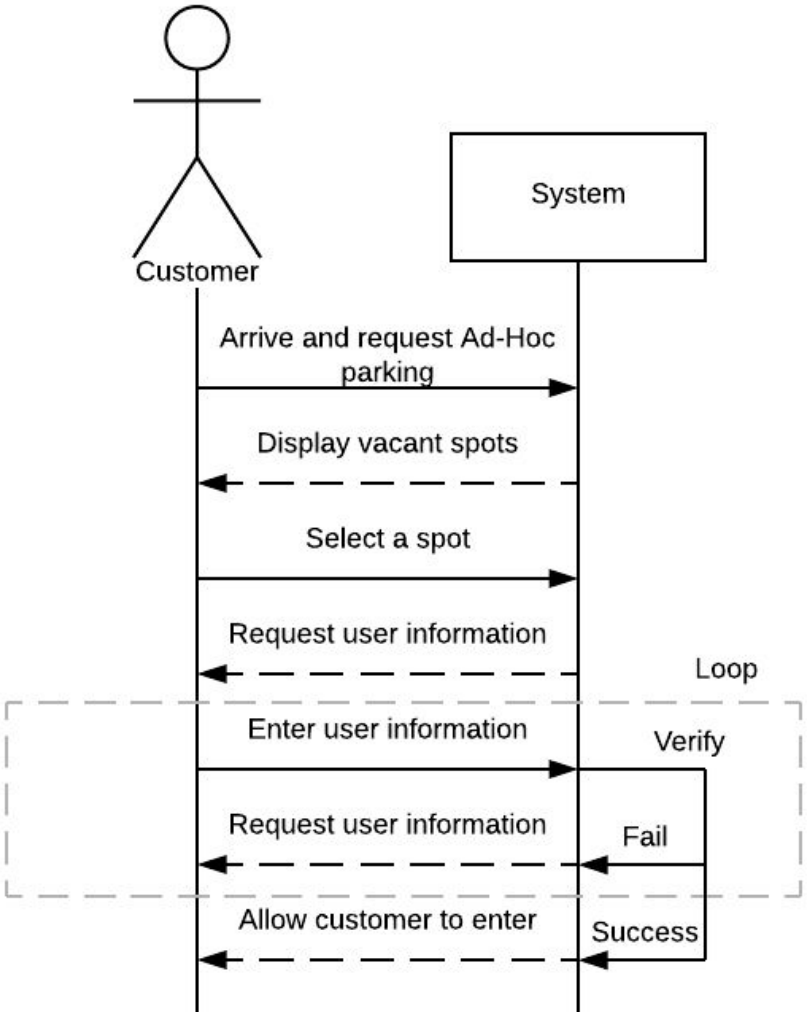


Use Case UC-3: Reservation
Related Requirements: REQ-03, REQ-04, REQ-06, REQ-13, REQ-15, REQ-18, REQ-19, REQ-30, REQ-36, REQ-38, REQ-40, REQ-43
Initiating Actor: Customer
Actor's Goal: To reserve parking spots at the garage in advance
Participating Actors: Customer, System
Preconditions: The spots that are being reserved must be reserved at least a day in advance
Postconditions: The customer's reservation will be stored in the database.
Flow of Events for Main Success Scenario: ->1. Customer logs into his account and clicks "Reserve a Spot". a. As a result of a failed login, the customer repeats step 1. b. If the login is successful, continue to step 2. <-2. The system returns a page that requests the time block for the customer's reservation ->3. The customer enters information into the required data fields (date, time block) <-4. The system checks the database of reserved spots during that time frame a. If there are no available reservation spots during the specified time frame, the system will inform the customer that there are no available spots to be reserved. (no reservation is made: flow ends here) b. If there are available spots during the desired time frame, the customer will be presented with a UI of the available spots that they can select from.(continue to step 5) ->5. The customer selects a parking spot to reserve <-6. The system enters the reservation into the reservation database. A message is displayed to show that the reservation went through.



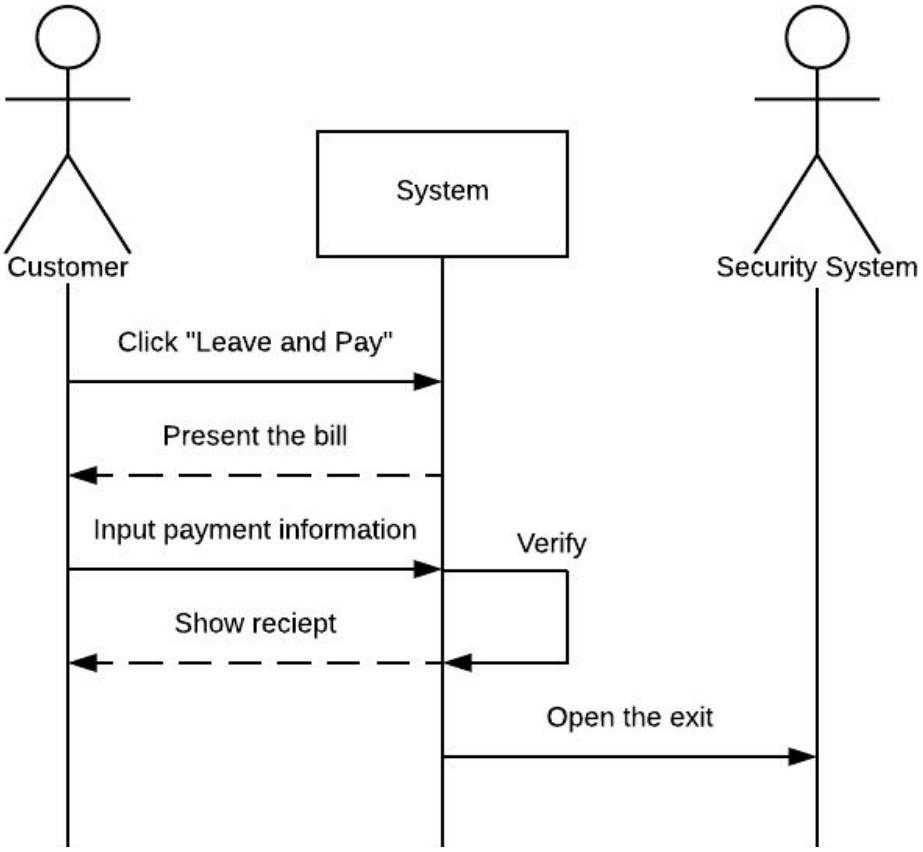
Use Case UC-4: Ad-hoc
Related Requirements: REQ-03, REQ-04, REQ-07, REQ-09, REQ-24, REQ-33, REQ-34
Initiating Actor: Customer
Actor's Goal: To obtain a parking spot at the garage without any reservation
Participating Actors: Customer, System
Preconditions: N/A
Postconditions: The customer will be able to park in the garage
Flow of Events for Main Success Scenario: ->1. The customer arrives at the parking garage and chooses the "Ad-hoc" option of parking. <-2. The system displays a real-time interface of the current parking situation of the garage (vacant and occupied spots) a. If there are no vacant spots, the customer has to leave b. If there are vacant spots, the customer can continue to step 3. ->3. The customer can select a vacant spot to park in. <-4. The system requests some basic information from the customer (name, license plate, contact information, and credit card information) ->5. The customer enters all the necessary information. <-6. The system checks the validity of the information. a. If the information does not fit the requirements (specifically credit card number), loop back to step 6. b. If the information fits the requirements, the system enters the customer's information

into the database. The customer's spot now shows up as "occupied" on the real-time interface, and the customer can proceed with the parking process.



Use Case UC-14: Payment
Related Requirements: REQ-03, REQ-04, REQ-06, REQ-13, REQ-15, REQ-18, REQ-19, REQ-30, REQ-36, REQ-38, REQ-40, REQ-43
Initiating Actor: Customer
Actor's Goal: To pay the bill based on the system information.
Participating Actors: The security system
Preconditions: The customers registered and entered to park the car.
Postconditions: The security system will let the customers out.

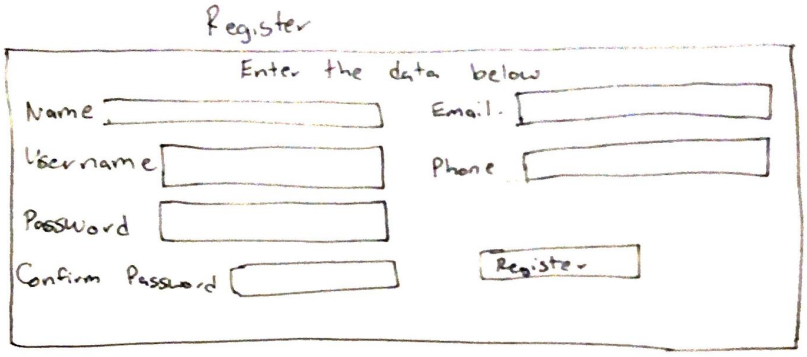
Flow of Events for Main Success Scenario:
->1. Customer logs into his account and clicks “Leave and Pay”.
    c. As a result of a failed login, the customer repeats step 1.
    d. If the login is successful, continue to step 2.
<-2. The system returns a page that shows the total money to be paid.
->3. The customer enters information into the required data fields (date, time block)
<-4. The system check the payment information and deduct it from the bank.
    c. If failure, return to 2
    d. If success, turn to 5.
->5. The customer receives the receipt.
<-6. The secure system will let the customer out.



3. User Interface Specification

3.1 Register - UC 1

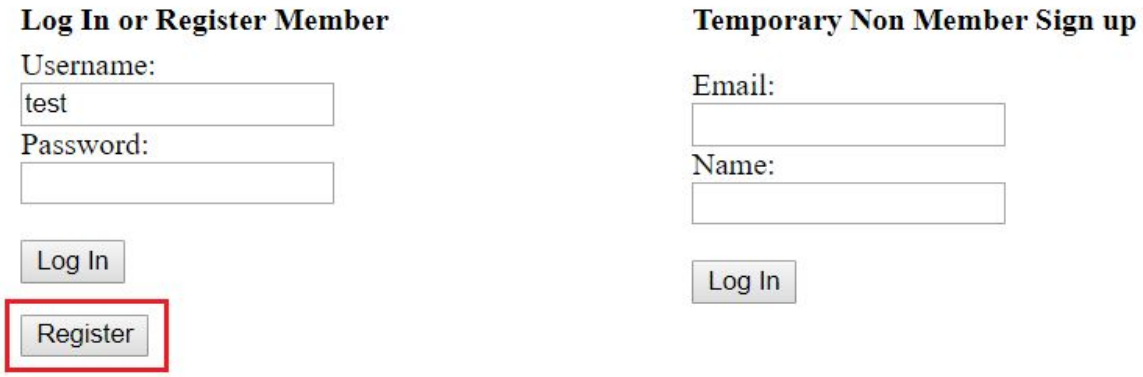
3.1.1 Preliminary Design



3.1.2 Effort Estimation

Navigation: total 1 click, as follows

a. Select Register as new member on the landing page shown below



Data Entry: total is indeterminant as keystroke per user will be different, 7 clicks

a. On Register page enter the follow fields and click to next field
    i. Name
    ii. Username
    iii. Email
    iv. Password

v. Confirm Password
vi. Phone

b. Select the submit button after all fields are entered

Mock up of Registration page with data entered

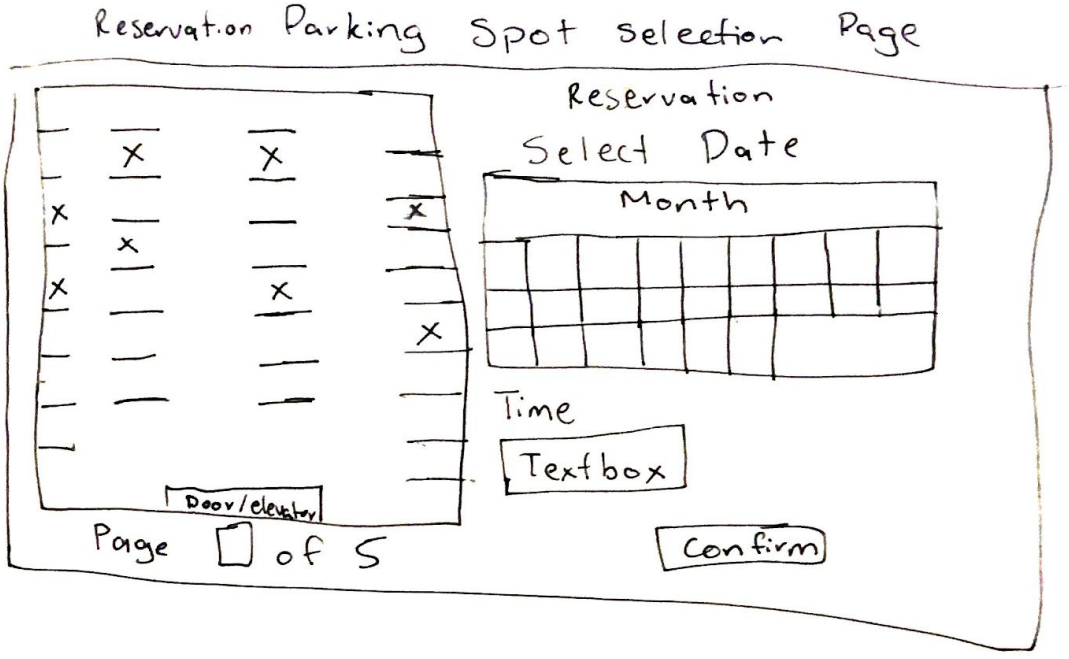
Registration Page

Please fill out all the fields below

Name John Doe
Email jd@gmail.com
Phone 1234567890
Password •••••••
Confirm Password •••••••••
Submit

3.2 Reservation - UC 3

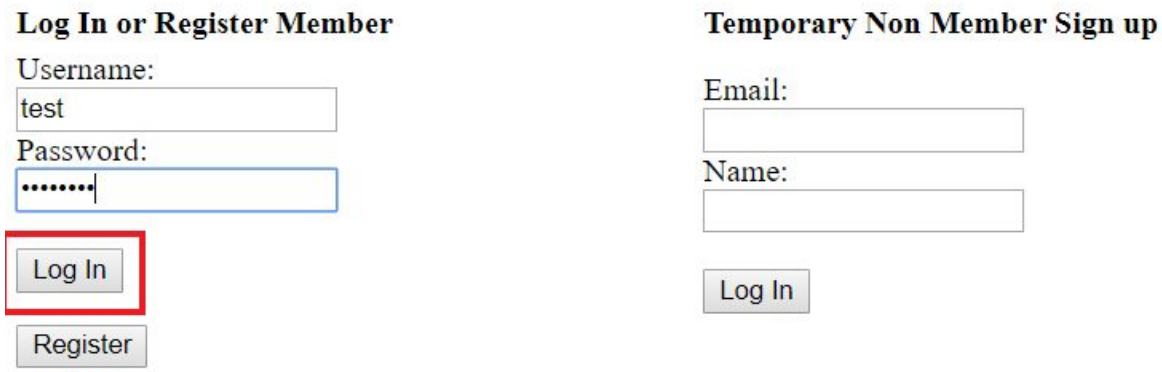
3.2.1 Preliminary Design



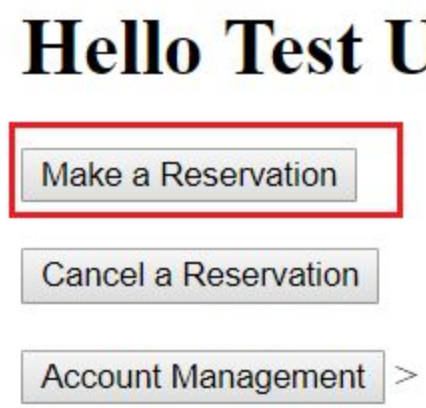
3.2.2 Effort Estimation

Navigation: total 2 clicks, indeterminant keystrokes as follows

a. Enter Login information as follows
    i. Enter Username
    ii. Enter Password
b. Click Login button (server sends user to account main page) shown below



c. On account page click Make a Reservation Page(user sent to make a reservation page) shown below

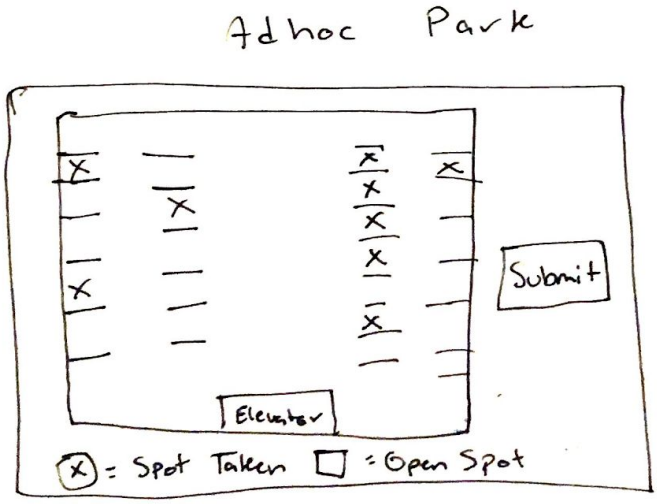


Data Entry: total is 5 keystrokes, 2 - 7 clicks

a. Select date of reservation by using calendar tool (number of clicks depends on date)
b. Enter time of reservation in 24 hour time (5 key strokes → xx:xx)
c. Select parking spot choice from the parking garage spot map tool

3.3 Ad - Hoc - UC 4

3.3.1 Preliminary Design



3.3.2 Effort Estimation

Navigation: total 2 clicks, indeterminant keystrokes as follows

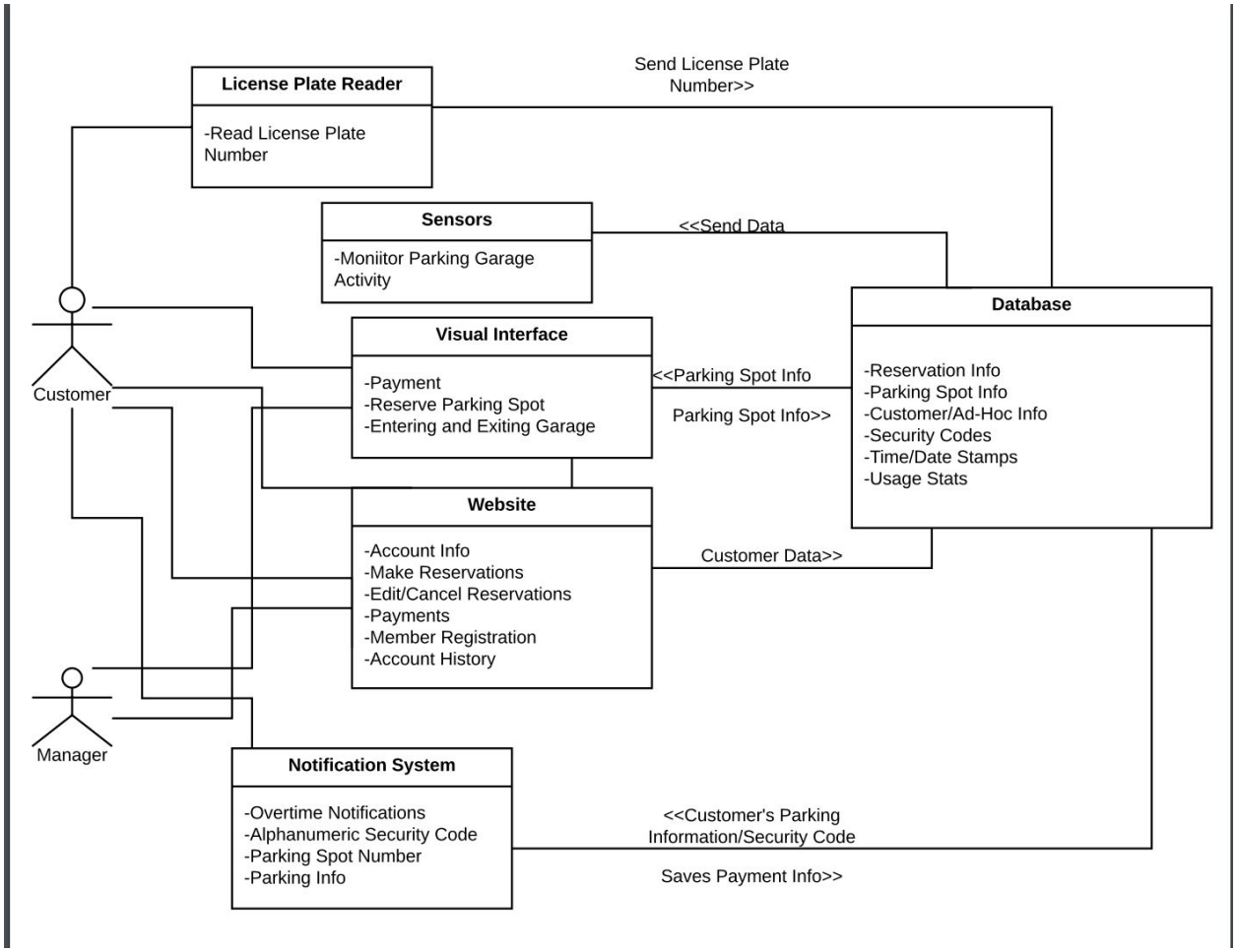
b. Enter Login information as follows
    i. Enter Username
    ii. Enter Password
d. Click Login button (server sends user to account main page)
e. On account page click Make a Reservation Page(user sent to make a reservation page)

Data Entry: total is 5 keystrokes, 2 - 7 clicks

d. Select date of reservation by using calendar tool (number of clicks depends on date)
e. Enter time of reservation in 24 hour time (5 key strokes → xx:xx)
f. Select parking spot choice from the parking garage spot map tool

4. Domain Model Analysis

4.1 Domain Model



4.2 Domain Model Derivation

The following domain model was derived from the use cases and requirements with the highest priorities. These use cases include allowing customers to login and/or register for an account (UC-1) and make reservations (UC-3), along with allowing both registered members and ad-hoc customers to view spots, park their car, keep track of their parking duration, and pay for their bill at the end (UC-2,UC-4,UC-5, UC-14, and UC-15).

The key boundary concepts of this system are the license plate reader, the visual interface that the customers will interact with upon entering the garage, the website, and the notification system. The key internal concepts of this system are the databased and sensors.
When the customer first enters the garage, the license plate reader takes a photo of the license plate and compares it to the database entries to see determine if the customer has an existing reservation. Based on that, the visual interface, which retrieves the data from the database, either allows the customer to select a parking spot in the garage or simply directs the customer to their reserved parking spot. The information inputted from the visual interface is saved in the database. Customers also have direct access to the website, where they can create accounts to make reservations and track their account history. Customer data along with any upcoming reservations are saved in the database. Along with this, on the website, customers will be able to see a display of all the available parking spots at a given time via the visual interface and database. The notification system periodically sends customers updates and notifications. Through the notification system, the customers can track the time that has elapsed and will also receive an alphanumeric security code that will be required for them to enter the garage. The notification system interacts with the database to retrieve customer and parking information. There are also sensors in place to monitor traffic activity in the garage and ensure that the the customers are parking in the correct spot.

4.3 Traceability Matrix

PW	Use Cases	License Plate Reader	Visual Interface	Website	Database	Notification System	Sensors
8	UC1			X	X		
15	UC2		X		X		X
23	UC3			X	X	X	
13	UC4		X		X		
17	UC5	X	X			X	X
3	UC6		X		X		X
13	UC7			X	X		
7	UC8					X	X
10	UC9	X	X			X	X
4	UC10					X	X
12	UC11			X	X		
7	UC12			X	X		
14	UC13		X	X	X	X	
23	UC14			X	X		
4	UC15			X	X	X	X
10	UC16		X	X		X	

4.4 Concept Definitions

Responsibility Description	Type	Concept Name
To check if customer has a reservation	K	License Plate Reader
To collect information for ad-hoc customers	D	Visual Interface
To give real time display of all currently available parking spots	N	Visual Interface
To allow ad-hoc customers to pay for parking	D	Visual Interface
To give managers access to current situation in parking spot	N	Visual Interface
To collect user information and make reservations	N	Website
To give registered customers an alternate payment method	D	Website
To register new members	N	Website
To manage smart pricing system	D	Database
To hold customer/parking/license plate/reservation info	N	Database
To keep track of time elapsed	D	Database
To manage usage stats	D	Database
To extract reservation information for incoming customers	K	Database
To allow managers to edit pricing and access customer info	K	Database
To send customers periodic notifications regarding time elapsed	D	Notification System

To send customers parking confirmation and alphanumeric security code	D	Notification System
To manage traffic control	N	Sensors
To ensure that customers are parking in the correct spots	K	Sensors

Association Definitions

Concept Pair	Association Description	Association Name
License Plate Reader ↔ Database	License plate reader scans the plate of cars and sends the plate number to the database where it is stored to be accessed later.	Store license
Visual Interface ↔ Database	Visual interface requests information on the status of parking spots from the database to be displayed to customers.	Get parking info
Website ↔ Database	Website sends information about reservations, users, and payment to the database to be accessed by other systems.	Update user info
Notification System ↔ Database	Notification system accesses database for customer information, such as phone number/email and parking spot number	Get notification info
Security System ↔ Notification System	Security system signals notification system to send authentication codes to customer	Allow security code
Security System ↔ Database	Security system requests security information stored in the database, such as a customer's unique security code	Get security code

4.5 System Operation Contracts

UC-1: Register

- Preconditions:
    - Customer is not already a registered member
    - Customer will enter a valid username, email address, and driver's license number
    - Customer will create a password
- Postconditions:
    - Customer's account information is stored in the database

UC-3: Reservation

- Preconditions:
    - Customer is logged into their account on the website
    - Reservations must be made at least a day before requested reservation time
    - The selected spot is available for reservation at the requested time
- Postconditions:
    - Customer's reservation will be stored in the database

UC-4: Ad-Hoc

- Preconditions:
    - Customer does not have a reservation for the current time
- Postconditions:
    - Customer is able to park in the garage
    - Customer will be registered for an account if they did not previously have one

UC-14: Payment

- Preconditions:
    - Customer is done parking and has moved their car from their spot
    - Customer had either parked by reservation or ad-hoc
- Postconditions:
    - Customer will be charged for their stay based on agreed upon rates
    - Security system will allow customer to leave

4.6 Mathematical Models

Here we mainly leverage the model and algorithm from [2]. We will modify the algorithm to work well in our use case. It is assumed that one driver only books one spot each time and group reservation is not considered in this work. Drivers can request reservation anytime during service. The reservation will be confirmed as long as there are available spots, which follows the assumption as in [1] that drivers care more about parking availability than spending time to figure out how to save money. Furthermore, since drivers' decision strategy is unknown to us, this work won't consider such factors and this handling strategy can also be found in similar works like [3]. Given assumptions above, we would like to adopt a mathematical model for our parking garage as presented in [2] which is defined as follows.

Let C denote the number of parking spots and each spot can serve N periods. X^t is a N-dimensional vector containing number of available spots for each period at booking time t, where $0 <= X_i^t <= C$ and i=1, …, N. Number of vacant spots can be utilized to help determine the price. For example, we could adjust to lower price when there are more available spots while to higher price when there are less. Thus, It is natural to define the N-dimensional pricing vector P^t that relies on X^t. On the other hand, drivers' demand can be easily affected by the price. Let's define a function $D^t_{[u,v]}(E[P^t_{[u,v]}])$ to represent the demand from u-th period to v-th period at booking time t, which mainly relies on the corresponding expected price. For simplicity, $E[P^t_{[u,v]}] = [\sum_{i=u}^{v} P^t_i]/(v-u+1)$ as presented in the [2].

Assume that reservation can be modeled by a poisson process. The intensity can be defined as $\lambda^t_{[u,v]}(X^t, P^t) = A^t_{[u,v]}(X^t, P^t)$, where $A^t_{[u,v]}$ is either 0 or 1 to indicate if there exists at least one available parking spot from u-th period to v-th period at booking time t. Thus, the total reservation rate is given by

$$\Lambda(X^t, P^t) = \sum_{u=1}^{N} \sum_{v=u}^{min\{N,\, u+n\}} \lambda^t_{[u,v]}(X^t, P^t),$$

where the n means maximum periods that a driver can choose. If there isn't, it could be simply set as N. As a result, our objective function is defined as following.

$$V(X^t,\ t) = max\{Q^t_1 V(X^t,\ t-dt) + Q^t_0[\sum_{i=u}^{v} P^t_i + V(X^t - e_{[u,v]},\ t-dt)]\},$$

where V is the maximum revenue from 0 to booking time t, dt is sufficient small during which at most one booking reservation arrives, Q^t_1 means the probability with one request while

$Q_0^t = 1 - Q_1^t$ means the probability with none and $e_{[u,v]}$ is a zero vector with u-th and v-th entry set as one.

$$V\left(x^t,t\right) = V\left(x^{t-\delta},t-\delta\right) + \max_{p^t \in P}\left\{\Lambda^t\left(x^t,P^t\right)\delta\left(\sum_{i=u}^{v} p_i^t - V\left(x^{t-\delta},t-\delta\right) + V\left(x^{t-\delta} - e_{u,v},t-\delta\right)\right)\right\}$$

Therefore, as $\delta\ (dt) \to 0$,

$$\frac{\partial V\left(x^t,t\right)}{\partial t} = \max_{p^t \in P}\left\{\Lambda^t\left(x^t,P^t\right)\left(\sum_{i=u}^{v} p_i^t - V\left(x^t,t\right) + V\left(x^t - e_{u,v},t\right)\right)\right\}$$

Optimality conditions

1. Basic case: n=1 (single period model)

// the below part is my part starting from 3.1.1. Please delete the commons after you finish writing your part.

We convert the demand function into the form $\lambda_t = \lambda(p_t) = ae^{-p_t}$ in which t could be omitted to simplify the expression and so we could get a new equation from Formula 3, which is listed below:

$f(x,p) = ae^{-p}(p - V(x,t) + V(x-1,t))$

To get the final price based on the time and parking lot condition, we calculate the first order condition $\frac{\partial f(x,p)}{\partial p} = 0$. And the optimized price is :

$p^*(x,t) = 1 + V(x,t) - V(x-1,t)$

We could also convert the demand function with a linear demand function, which is $\lambda(p) = a - b \cdot p$. Here we use a to represent the base demand and b to represents the price sensitivity factor. So we rewrite the function as following:

$f(x,p) = (a - bp)(p - V(x,t) + V(x-1,t))$

To get the final price based on the time and parking lot condition, we calculate the first order condition $\frac{\partial f(x,p)}{\partial p} = 0$. And the optimized price is :

$p^*(x,t) = \frac{a+b(V(x,t)-V(x-1,t))}{2b}$

2. Multiple-period model

There are also use cases where the user reserve one parking space for n periods (here n > 1). Here we also assume that the user could reserve for less than the service horizon N. So we could obtain $(N+1)N/2$ classes in this case. For example 1-1, 1-2,...,1-N,2-2,2-3,...2-N,....,N-N. Here we focus on the linear demand function and so we have :

$$\lambda^t_{[u,v]}(X^t, P^t) = A^t_{[u,v]}(X^t, P^t)(a^t_{[u,v]} - b^t_{[u,v]} \times \sum_{i=u}^{v} p^t_i/(u - v + 1))$$

Here, we use $a^t_{[u,v]}$ to represent base demand of class u-v at time t and $b^t_{[u,v]}$ to represent the price sensitivity factor of class u-v at time t related to the current average price. And so we can change the form of equation as following:

$$f_t(x^t, p^t) = max\{\sum_{u=1}^{N}\sum_{v=u}^{N} A^t_{[u,v]}(X^t, P^t)(a^t_{[u,v]} - b^t_{[u,v]} \times \sum_{i=u}^{v} p^t_i/(u - v + 1))(\sum_{i=u}^{v} p^t_i - \Delta V_{[u,v]}(x^t, t))\}$$

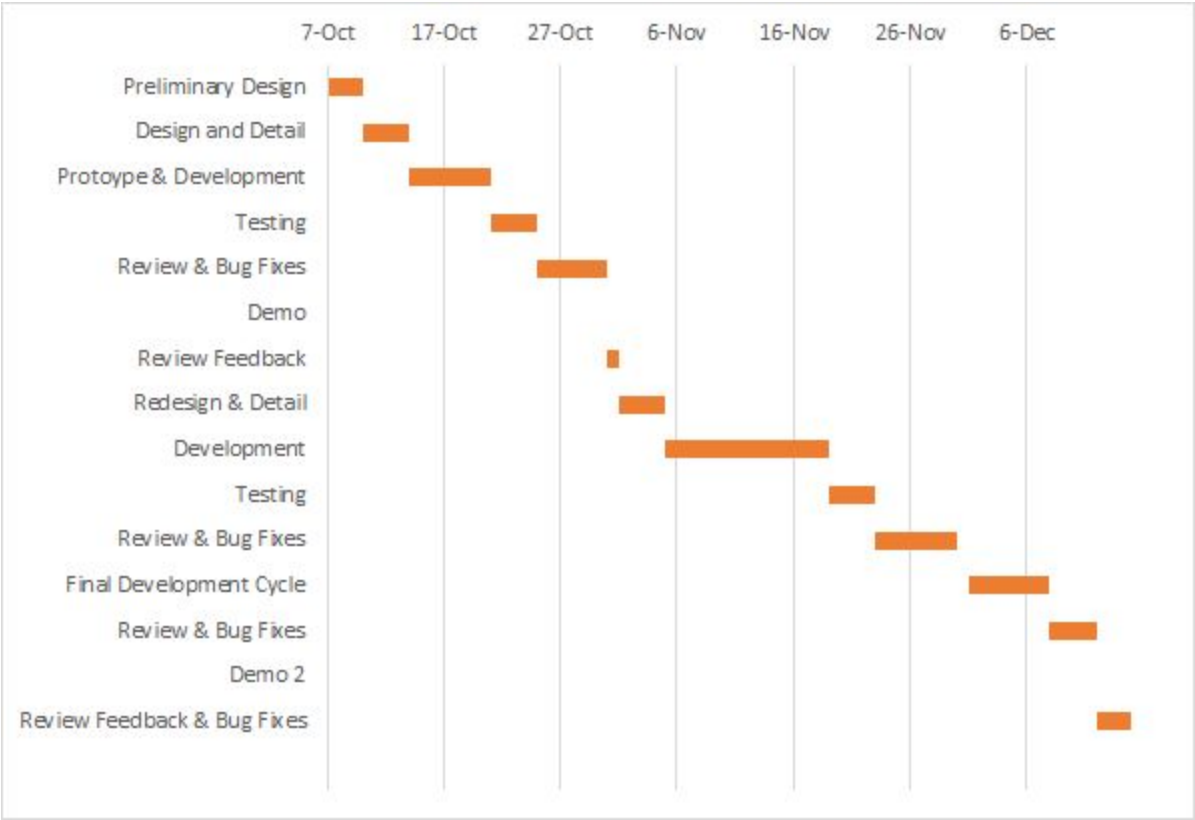
To get the final price based on the time and parking lot condition, we calculate the first order condition $\frac{\partial f(x,p)}{\partial p} = 0$.

4.7 Plan of Work

Below is a task list and Gantt diagram showing the estimated start date, duration, and end date of the upcoming tasks for the project. It is obvious that the task flow is heavily modeled off of the software development lifecycle. Descriptions of each of the tasks are also listed below.

task	start date	duration	end date
Preliminary Design	7-Oct	3	10-Oct
Design and Detail	10-Oct	4	14-Oct
Prototype & Development	14-Oct	7	21-Oct
Testing	21-Oct	4	25-Oct
Review & Bug Fixes	25-Oct	6	31-Oct
Demo	31-Oct	0	31-Oct
Review Feedback	31-Oct	1	1-Nov
Redesign & Detail	1-Nov	4	5-Nov
Development	5-Nov	14	19-Nov
Testing	19-Nov	4	23-Nov
Review & Bug Fixes	23-Nov	7	30-Nov
Final Development Cycle	1-Dec	7	8-Dec
Review & Bug Fixes	8-Dec	4	12-Dec
Demo 2	12-Dec	0	12-Dec
Review Feedback & Bug Fixes	12-Dec	3	15-Dec

Note that in the Gantt diagram below Demo 1 & 2 are tasks that are only one day therefore they will have 0 duration and will not have a visual “bar” for a duration.



Task	Description
Preliminary Design	-Design basic UI for each page -Establish routes for each user case (determine which page goes to where) -Beginning dev of math model for smart pricing system -Determine Mongo DB collections and objects that will be required for each subteam
Design & Detail	-Finalize UI for each page (HTML & CSS) -Determine user page interaction flow for all possible use cases -Research and simulation of math model for smart pricing system -Finalize Object variables for each module -Describe in detail module interaction and data necessities for each one
Prototype & Development	Systems: (Detailed in previous sections)

	1. Entrance Gate a. Detect car at the entrance b. REQ 1, REQ 2 c. Allow user in after parking spot choice 2. Parking Spot Choice a. REQ 3 b. Send spot data to elevator system c. Send open signal to entrance gate system 3. Elevator a. Receive spot choice data and bring user to correct floor 4. Ad Hoc Parking Detection a. Detect user spot choice if choice has changed b. Relay new spot choice to Parking spot choice module 5. Spot Detection/Verification a. Detect car is in spot b. Relay spot choice for adhoc and spot check for other cases 6. Traffic Management a. Control elevator and only allow 1 person per level to park b. Receive spot verification from from other module and allow next person to park on the floor 7. Email Notification System a. Send email to users if the parked in the incorrect spot b. Send code to allow user to exit/enter spot 8. Security System 9. Pricing System
Testing	-Allowed other team members to code review and perform QA testing -Other team members will list any bugs and issues with code -Other team members will give recommendations as well

Review & Bug Fixes	-review QA suggestions and bugs report -fix bugs or detail systems interactions that are causing bugs

4.8 Product Ownership

Siddharth

Done	In Progress	Future Responsibilities
- On Screen Appearance Requirements - User Interface Specification - Database and Project Report management - Log In authentication -All page routing	-Reservation Form -Spot Choice UI and Logic -User sessions for persistent log in	-Ad Hoc spot verification -Spot Detection and Verification

Ridhima

Done	In Progress	Future Responsibilities
-Enumerated Functional/Non Function Specifications -Use Cases -Traceability Matrices -Domain Model Analysis -Concept Definitions	-Registration Form -Login UI	-Security/Notification System -Visual Interface (Payment, Ad-hoc customer info)

Corey Chen

Done	In Progress	Future Responsibilities
-Enumerated Functional/Non Function Specifications -Use Cases -Stakeholders -Actors and Goals -Fully Dressed Use Cases -Customer Statement of Requirements	-Registration Form -Login UI/Page	-Elevator Module -Security/Notification System

Siyu Liao

Done	In Progress	Future Responsibilities
-Enumerated Functional/Non Function Specifications -Use Cases -Actors and Goals -Fully Dressed Use Cases -Customer Statement of Requirements -Novelty Declaration	- Mathematical Modeling - Input Simulation - Model Simulation	- Numerical experiments - Model analysis - Exploring machine learning from real world data

Chunhua Deng

Done	In Progress	Future Responsibilities
-Enumerated Functional/Non Function Specifications -Use Cases -Actors and Goals -Fully Dressed Use Cases -Customer Statement of Requirements -Novelty Declaration	- Mathematical Modeling - Previous work analysis - Model Simulation	- Numerical experiments - Smart Charging module - Machine learning from real world data

Xianglong Feng

Done	In Progress	Future Responsibilities
-Enumerated Functional/Non Function Specifications	- Mathematical Modeling - Model Simulation	- Model analysis - Machine learning from real

-Use Cases -Actors and Goals -Fully Dressed Use Cases -Customer Statement of Requirements -Novelty Declaration	- Data Collection	world data - Model Fusion - Model Verification

Jonathan Garner

Done	In Progress	Future Responsibilities
-Enumerated Functional/non Functional Specifications -Use Cases -Actors and Goals -Customer Statement of Requirements -Domain Model Analysis	-Reservation Form -Spot Choice UI and Logic -User sessions for persistent log in	-Ad Hoc spot verification -Spot Detection and Verification

Reference

[1] Gallego, Guillermo, and Garrett Van Ryzin. "Optimal dynamic pricing of inventories with stochastic demand over finite horizons." *Management science* 40.8 (1994): 999-1020.

[2] Tian, Qiong, et al. "Dynamic pricing for reservation-based parking system: A revenue management method." *Transport Policy* 71 (2018): 36-44.

[3] Lam, William HK, et al. "Modeling time-dependent travel choice problems in road networks with multiple user classes and multiple parking facilities." *Transportation Research Part B: Methodological* 40.5 (2006): 368-395.