Software Engineering Project

ONLINE CAR PARKING RESERVATION SYSTEM

Group-5

Members

No.	Name	ID
1	Harshal Markana	201801143
2	Ishang Kumar	201801071
3	Raj Mahla	201801243
4	Darshan Prajapati	201801146
5	Jenil Khandhara	201801217
6	Sambhav Agrawal	201801063
7	Nipun Patel	201801234
8	Parthiv Patel	201801463
9	Mahi Patel	201801039
10	Sudiksha Thusu	201801469

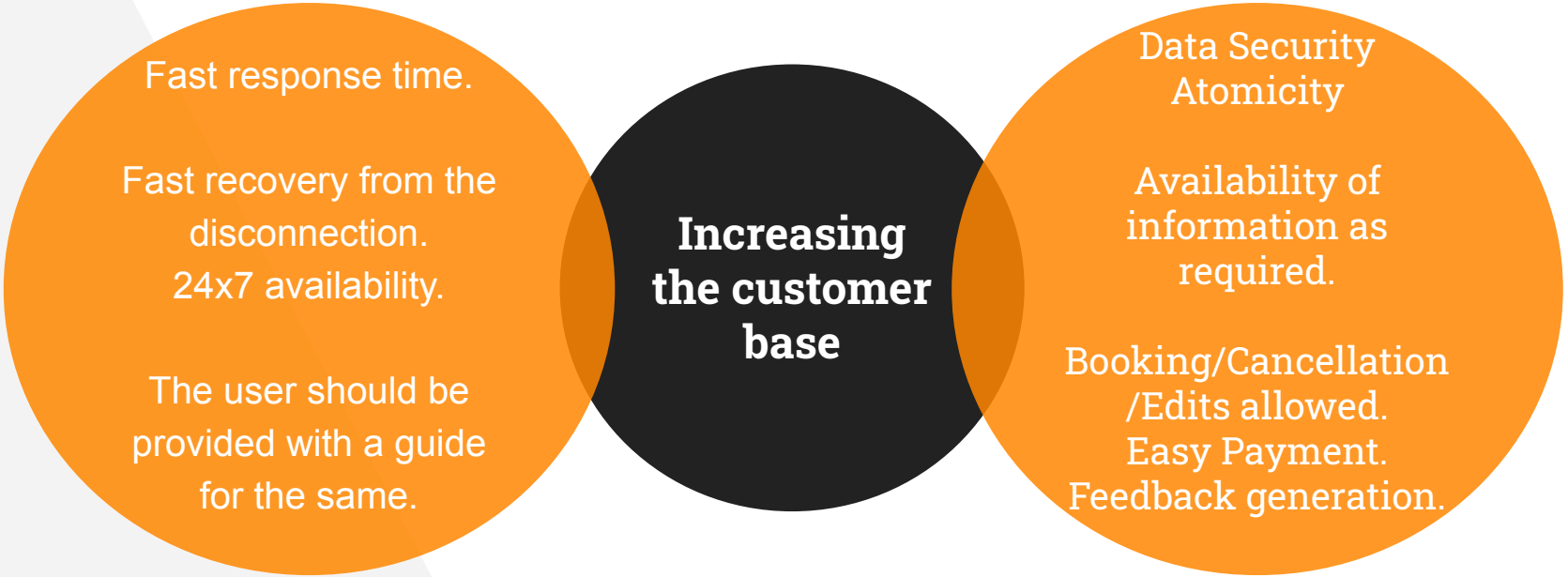
Problem Statement

Currently, most of the existing car parking systems are manually managed and a little inefficient. In urban areas, where the number of vehicles is higher as compared to the availability of parking spaces, a lot of time being wasted in searching for parking locations. Hence online booking parking system is a proposed method that users can reserve their parking places using the web. Providing a simple web application for parking vehicles. Booking for a parking slot at home. Can search nearby places using Google map. Easy payment system. Parking owners can add their own parking places. Make it easy to automate parking owners and customers.

Objectives

- Our main objective was to create an online car parking reservation system to improve the current mismanagement in allocating a parking area for all vehicles
- This system will enable drivers to locate and reserve a parking place online and on confirmed payment, the user will get an e-receipt via email. The users will also be able to see their previous booking and can add review to the parking areas.
- The parking owner will also be allowed to login into the system and add the parking areas. They will be able to see the current parkings in his area and will have an access to view the feedback of customers.

What we aim for?



Milestones

- Requirement elicitation:

  a. Brainstorming upon different ideas that need to be implemented

  b. Researching and analysing the available resources and websites.

  c. Interviewing the potential users of the system

  d. Deciding the model we need to work on for project

  e. Analysing functional and non-functional requirements

- Design Thinking:

  a. Use case formation and flow of the project decided

  b. Based on the same use case diagram, activity diagram, sequence diagram, state diagram and class diagram were formed.

  c. User Interfaces were designed firstly the wireframes and then the actual UI design

Milestones

- Implementation:

  a. Team was divided into 2: Frontend and Backend

  b. The Backend team used NodeJS for backend and MongoDB for data management. We looked upon the implementation of search via maps, to create a payment system and generate an online receipt. Along with that we created an review system too.

  c. The basic input forms and display pages were made as ejs files

  d. The frontend team worked upon deciding the website theme and layout.

  e. Later the styling to the ejs files were done by adding CSS.

Milestones

- Testing:
  a. Checked upon the all the features
  b. Modified the responsiveness of the website
  c. Identified the missing parts
  d. Implemented black box testing to identify the bugs
  e. Analysed the boundary cases

Contributions

No.	Name	Work done
1	Harshal Markana	● Google Authentication features for user ● Update feature for User and owner ● The new schema for parking slots to include bike and car prizes and the number of slots. ● Responsiveness of page user_index, date and time entry, feedback page ● Sequence Diagram
2	Ishang Kumar	● Added mailing feature after payment confirmation. ● Added payment ejs file for the payment details. ● UI design of payment page which lets the user enter his personal and credit card details and email address to which the invoice is to be sent. ● Added HTML and CSS for the invoice sent(mail) to the user on payment confirmation. ● Responsiveness of Login/Signup and Map. ● Sequence Diagram.

Contributions

No.	Name	Work done
3	Darshan Prajapati	● Added schema for parking location. ● Added cluster map connected with the database ● Added popup on the map for navigating from map to booking and feedback page. ● Users can search location and see their current location on the map. ● Added schema for reviews. ● Added reviews and rating page for users to give feedback for each parking location and see the reviews added by other users. ● Added feature of deleting users' own reviews. ● Added average rating attribute which updates after create and delete the review. ● Added the feedback and rating view page for his parking location given by users. ● Added redirection messages for user-related pages. ● Added map on feedback page for all parking locations with location marker. ● Black Box Testing ● Class Diagram

Contributions

No.	Name	Work done
4	Jenil Khandhara	• Initialized express app and created schema for users and owners. • Added register and login functionality for two different types of users i.e. owner and user using passport and JSON web token. • Changed routes according to rest notation • Added schema to store the slots booked by the user • Added functionality for users to see his/her details of current and previous bookings using a database query. • Added the customer status page where the owner can see the details of users who have parked on their parking location • Added redirection messages for login, register, and owner related pages • Added dynamic data of slot booked by the user in the email sent after payment • Black Box Testing • Class Diagram • Deployment of application on Heroku

Contributions

No.	Name	Work done
5	Sambhav Agrawal	• Added ejs file for the entry of date and time. • Made database query for finding empty slots of parking location for the date and time entered by the user. • Solved bug that prevents the user to enter previous date and time in accordance with the selected date. • State diagram
6	Sudiksha Thusu	• Designing Booking, Payment Details Page, and Homepage. • Homepage HTML and CSS which includes a title page, login, and register, the second part is the information about the website, and the third part is the feedback about the website from the user. • Activity Diagram • Black Box Testing
7	Raj Mahla	• UI Designs for Login, Signup, Booking, and Payment Details and Feedback page. • State diagram • Homepage HTML and CSS, which includes the title page, login, and register, part contains information about the website, last is about the feedback and queries.. • Feedback Page CSS, which shows feedback of all users and average rating, and the users can add feedback for particular. • Manual GUI testing if the website.

Contributions

No.	Name	Work done
8	Nipun Patel	• UI designs for Login, Signup, Feedback page, and HomePage. • State diagram • Added Homepage HTML and CSS, which includes the title page, log in, and register, Second part is the information about the website, and Third part is feedback about the website from the user. • Added Feedback Page CSS, which shows feedback of all users and average rating. Users can add feedback for particular parking in the feedback form.
9	Parthiv Patel	• Created the UI design for the Parking slot Page, Details of Parking, Parking Owner. • Added the CSS styles for login,signup and view booking history page where user can see his/her bookings. • Sequence Diagram • Static analysis testing of CSS files • Responsiveness of Update owner, Update user and view booking history pages. • Manual GUI testing if the website.

Contributions

No.	Name	Work done
10	Mahi Patel	• Create the UI design for the Parking slot Page, Details of Parking, Parking Owner. • Added CSS styles for the: • Login, Register page, • User Index page where parking slots can be viewed, • Booking a slots page where the user can book the slot according to the date and time provided, • Update user and owner account page. • Owner dashboard where the owner can see all the parking areas under him, • Add a parking area page where the owner can add a new parking slot and also the parking status page where the owner can see about the users currently parked in the area and their reviews. • Created responsive login and signup page • Created activity diagram • Black box testing

Mistakes

1. Our team started thinking about the code and implementation without requirement elicitation, so then we identified all the functional and nonfunctional requirements and with the help of diagram, we were able to get a better picture.
2. Use of github was not wisely done in earlier stages but then the team learnt and handling the changes was much easier.
3. Sometimes miscommunications lead to bad implementation decisions.

Mistakes

Frontend:

1. Initially our frontend design was static but due to varying screen size we tried to make it responsive.
2. In the beginning the frontend and backend teams were working independently, so initially we faced many merge conflicts. To reduce them we used ejs files for a basic input and then the fronted team added CSS to it.
3. We started coding for frontend pages without making any proper wireframes (web page designs)
4. In some of the pages, some of the style commands were assigned custom size values which created a lot of problems in alignments while checking the responsiveness.

Mistakes

- **Backend:**

1. Slot was booked after choosing date and time entry and it was solved by booking it after confirm payment.
2. Adding review was updating avg rating but deleting wasn’t
3. Date and time for past date were working.
4. After logout user was able to access page directly from the url.
5. Back button issue.
6. Google sign in button wasn’t disappearing for registration of owner.
7. Close button for flash message was not working.

Achievements

- We were able to implement an online car parking reservation system where the user can book a slot online.
- Our website focuses on both the user modules: Driver and Parking owner
- For the ease of searching the areas, we used MapBox api.
- After the user books a slot, we have created an e-payment system for cashless transaction and after confirmation an e-Receipt will be sent to the User email ID.
- As an owner one can add a parking area on the website and also can track the customers parked in that area.
- The user can also provide his/her feedback and which owner can easily see and improvise.
- We were able to work as a team and combine all the ideas and technologies to implement most of the functional and non-functional requirements.

Take Away

- We learnt that project is not just about writing code. Instead of taking a deep dive into implementation, all the phases of requirement elicitation and design thinking helped us to understand the model better.
- Through the help of use case model, activity diagram and sequence diagram, we got a clear idea of how the flow of the website should go, who are going to be the actors (users) and what each functioning will be.
- We learnt about the importance of github, how to use it including all the commands and the collaborations.
- In the implementation part too we learnt about various language tools- NodeJS, CSS, HTML and to add features like searching via a Map, sending e-Receipts via email etc.

Take Away

- We learnt to try different ways to make the website more user friendly by making it responsive and easy to understand.
- “Testing software” lectures, helped us to identify how we can find bugs in our system and make our website user friendly. Although due to time constraints, we were not able to do all the testings but we did perform black box testing, GUI testing
- And lastly, this project being a group project and online mode, there were a lot of possibilities for miscommunication, and not in the beginning but somewhere in between we did felt that, but we were able to overcome it soon, we held regular meetings and rectified each of the problems and work together.
- We learnt the lesson of team work and how can we utilize different thought to bring something productive

Rating of Software Artifacts (0-5)

- **Requirement set:**

  Requirement Elicitation: 3.5

  Functional and non-functional requirements:4.5

  Use case identification:5

  Use case diagram:5

Rating of Software Artifacts (0-5)

- **Design Set:**

  Activity diagram: 4

  Sequence diagram: 3

  State diagram: 3

  Class diagram: 5

  Wireframes: 3

  UI designs: 4.5

Rating of Software Artifacts (0-5)

- **Implementation Set:**

  Backend database: 4.5

  Frontend styling: 4

- **Implementation Set:**

  Responsiveness: 3

  Black box testing:4.5

  GUI testing: 3

  Static testing: 2

Overall rating of Project

Rating: 8/10

THANK YOU!

WEBSITE LINK: https://planyourpark.herokuapp.com/
GITHUB REPOSITORY LINK:
https://github.com/3005coolik/Group5-Online_Car_Parking_Reservation_System
DEMO VIDEO: https://youtu.be/rKykvggq0Xs