|  |
| --- |
| **ParkMe Project**  **CSC 4350 Fall 2023**  Group 7:  Jay Patel  Seth Quiros  Hieu Vu (Coordinator)  Wajeeha Mehr  Christopher Reed  Date of Submission: 09/??/2023 |

**SECTION 1:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Assignee Name** | **Email** | **Task** | **Duration (hours)** | **Dependency** | **Due date** | **Evaluation** |
| Jay Patel | jpatel182@student.gsu.edu | Problem Statement, Use cases, Requirements, Database |  | None |  |  |
| Seth Quiros | Squiros1@student.gsu.edu | Use cases, Requirements, Database |  | None |  |  |
| Hieu Vu  (Coordinator) | hvu13@student.gsu.edu | Use cases, Requirements, Database |  | None |  |  |
| Wajeeha Mehr | wmehr1@student.gsu.edu | Use cases, Requirements, Database |  | None |  |  |
| Christopher Reed | creed27@student.gsu.edu | Use cases, Requirements, Database |  | None |  |  |

**SECTION 2:**

**[Problem Statements]**

**SECTION 3:**

**Requirements:**

**[Diagram]**

**[Diagram]**

**Use Cases:**

Use case 4: Permit Purchase and Management

Actors: User, Admin, Database

Description:

* User logs in and accesses the dashboard.
* User selects "Permit Purchase and Management."
* User chooses a permit type (e.g., daily, monthly).
* System validates user's eligibility.
* User makes a secure payment.
* System records the transaction.
* User manages purchased permits (renew, extend, modify).
* Admins oversee permit data, types, availability, and pricing.

Exception Path: If the user's login fails, they will need to log in again as described in the "User Login" use case before proceeding with permit purchase and management.

Alternate Path: The main path includes the steps necessary for the user to purchase and manage parking permits. The process is straightforward unless there are errors related to the user's eligibility or payment processing.

Pre-condition: A valid user account must already be created in the system, and the user must be successfully logged in.

Post-condition: After successfully purchasing or managing parking permits, the user's permit information is updated in the system, and they can access their permit details through their web browser.

Use case 5: Interactive Campus Map

Actors: User, Database

Description:

* User accesses ParkMe Web Page and selects "Interactive Campus Map."
* System loads a detailed campus map with interactive features.
* Zoom and pan for exploration.
* Search bar to find specific locations.
* Filters for parking, buildings, dining, and more.
* Clickable icons for points of interest.
* Real-time data for parking and shuttle info.
* Users can search locations (e.g., "Library") and view details when clicking icons.
* Users get step-by-step directions with travel times.
* Map updates in real-time for parking, shuttle routes, and events.
* Users customize map view, save favorites, and set preferences.
* Help and support features include FAQs, live chat, or a hotline for assistance.

Exception Path: If there is a technical issue with loading the map or accessing real-time data, the user may receive a notification indicating that the map is temporarily unavailable. The user can try again later or contact support for assistance.

Alternate Path: The main path includes all the necessary steps for users to interact with the campus map effectively. However, users may choose to explore the map without using specific features like searching or getting directions, following their own navigation preferences.

Pre-condition: The user must have access to the ParkMe Web Page and an internet connection.

Post-condition: After using the interactive campus map, the user gains a better understanding of the campus layout, locates their desired destinations, and may plan their parking or transportation accordingly for a more efficient campus experience.

Use case 6: Parking Reservation

Actors: User, Admin, Database

Description:

* Users log in and access the user dashboard.
* In the dashboard, they choose "Parking Reservation."
* They select reservation details (lot, date, time).
* The system validates availability and proceeds to payment.
* Users pay securely and receive a confirmation.
* Reservation details are stored in the database.
* On the day, users use a QR code or reservation ID for parking.
* Admins monitor reservations and adjust parameters.
* Users can modify or cancel reservations per policies.
* The system sends reminders before reservation expiry for renewal or checkout.

Exception Path: If the selected parking lot does not have availability for the specified date and time, the system informs the user and suggests alternative options or times.

Alternate Path: The main path encompasses the necessary steps for users to make parking reservations. Users can customize their reservations based on their specific needs and preferences.

Pre-condition: The user must have a registered account on the ParkMe Web Page and be logged in.

Post-condition: After successfully completing a parking reservation, the user has a guaranteed parking spot for their specified date and time, improving their campus parking experience.

Use Case 7: Alternative Transportation Information

* Actors: Student, admin, database
* Description:
* The student logs into the ParkMe platform using their credentials.
* Once logged in, the student navigates to the "Transportation Information" section of the user dashboard.
* In the Transportation Information section, the student selects their desired date and time for commuting to campus.
* The system retrieves and displays available parking spots, shuttle schedules, and other transportation options on campus for the selected date and time.
* If the student finds suitable parking spots and shuttle schedules for their desired time, they can select and reserve a parking spot or note down the shuttle schedule.

Alternate Path: If there are no available parking spots or shuttle schedules for the desired date and time, the system informs the student and suggests alternative dates or times for commuting.

Exception Path: In case of a system error preventing the student from searching for transportation options, the system displays an error message and advises the student to try again later or contact technical support.

Pre-condition: The student needs to get to campus and wants to find parking or transportation options.

Post-condition: The student successfully views and selects from available parking spots and shuttle times, enabling them to plan their commute to campus efficiently.

Use Case 8: User Feedback and Support

* Actors: Student, admin, database
* Description:
* After using ParkMe services, such as parking reservations or shuttle schedule lookup, the student decides to provide feedback or requests technical support.
* The student accesses the "Feedback and Support" section on the ParkMe platform, typically found in their user dashboard.
* In this section, the student has the option to fill out a feedback form or submit a support request via an email or form.
* The student provides relevant details, which may include their name, contact information, a description of the issue, and any suggestions for improvement.
* If the student had issues with a parking reservation, they may also have the option to call the support team for a refund or compensation.

Alternate Path: If the student experienced significant issues with their parking reservation, such as being unable to use their reserved spot due to a system error, they may choose to call the support team for immediate assistance, refund, or compensation.

Exception Path: If the feedback form or support request fails to submit due to a technical error, the system displays an error message and advises the student to try submitting their feedback or request again later. Alternatively, the student may be provided with alternative contact information for support.

Pre-condition: The student recently used ParkMe services to commute to campus and has feedback or requires technical support.

Post-condition: The feedback or support request is successfully submitted to ParkMe for review and potential improvement of services. If applicable, the student's issue with a parking reservation is addressed, and any necessary refunds or compensation are provided.

Use Case 9: Admin Parking Management

Actors: Admin User, Student User, Database

Description: Allows an authorized admin user to make changes to the parking system.

* Admin user logs in and can access the admin dashboard.
* In dashboard, admin chooses to modify parking or create new parking
* If admin chooses “Modify Parking”, they can adjust parking lot information, adjusting pricing and managing reservations.
* If admin chooses “Create New Parking”, they can create a new parking area and input mandatory information (such as, lot number, how many available parking spots, location, etc.)
* The system validates the information, and it is inputted into the database.
* The student user receives notifications of the modifications.

Exception Path: Technical issues: such as server is down or database errors, the admin may not be able to make changes to the parking system. An error message will be displayed. And invalid data input, if the admin submits invalid data or leaves a field empty when making changes, there is an error message.

Pre-condition: Admin needs to have a valid account registered with the system, be logged into their account and have the necessary permission to make changes.

Post-condition: Any changes the admin makes are saved and reflected in the parking system. If there are any technical issues, the system is brought back to a stable state and errors are not saved.

Use Case 10: Admin Permit Management

Actors: Admin User, Student User, Database

Description: Allows an authorized admin user to make changes to permits within the parking system.

* Admin user log in and can access the admin dashboard.
* In the dashboard, admin chooses to modify the permit or create a new permit.
* If admin chooses “Modify Permit”, they can modify and revoke existing permits.
* If admin chooses “Create New Permit”, they can create a new parking area and input mandatory information (such as, lot number, location, etc.)
* The system validates the information, and it is inputted into the database.
* The student user receives notifications of the modifications.

Exception Path: Technical issues: such as server is down or database errors, the admin may not be able to make changes to the permit system. An error message will be displayed. And invalid data input, if the admin submits invalid data or leaves a field empty when making changes, there is an error message.

Pre-condition: Admin needs to have a valid account registered with the system, be logged into their account and have the necessary permission to make changes.

Post-condition: Any changes the admin makes are saved and reflected in the parking system. If there are any technical issues, the system is brought back to a stable state and errors are not saved.

Use case 11: Admin User Feedback Analysis

Actors: Student, Admin

Description: Feedback section for any user experiencing errors or problems within parkme.

Alternate Path: Contacting email if no response within 24 hours. Also, a help support option to assist your difficulties.

Exception Path: System does not allow feedback. Contact support or email admin for help.

Pre-condition: Student needs to be experiencing errors or glitches within the system.

Post-condition: User should be able to submit feedback on the website to fix errors or suggestions for future upgrades to the site.

Use case 12: Security and Privacy Management

Actors: Student, Security admin

Description: Student should be worry free from sensitive information leaking out to the public.

Alternate Path: Secure account with two factor Authentication.

Exception Path: Student does not want to take forever to log in. Student can access guest pass with lesser amount of sensitive information needed.

Pre-condition: Student wants information safe and not accessible to the public.

Post-condition: Student has the option to set up security on the account or can use guest section for less sensitive information needed.

**System Requirements:**

Requirement: 4

Use Case: 4

Name: Secure Payment Processing

Introduction: Ensure secure payment processing for parking permit purchases.

Rationale: To safeguard user financial information and provide a seamless payment experience.

Input: User selects permit type and provides payment details.

Requirement Description: Implement secure payment processing with encryption, payment gateway integration, clear error handling, and email receipts.

Output: Secure payment transactions, payment confirmation, email receipts, and transaction logs.

Test Cases: TBD

Requirement: 5

Use Case: 5

Name: Real-time Data Updates

Introduction: Ensure real-time data updates on the interactive campus map.

Rationale: To provide users with accurate and up-to-date information for effective navigation.

Input: User interacts with the map, selects filters, or searches for locations.

Requirement Description: Implement a system that regularly updates map data, including parking availability, shuttle locations, and events. Updates should occur at least every minute to ensure real-time accuracy.

Output: Users see the most current information on the map, such as parking availability and shuttle locations.

Test Cases: TBD

Requirement: 6

Use Case: 6

Name: Reservation Confirmation

Introduction: Provide users with reservation confirmations.

Rationale: To assure users that their parking reservations are successfully made.

Input: User selects reservation parameters and makes a payment.

Requirement Description: After a successful reservation, the system must generate a confirmation message with the reservation details, including date, time, parking lot, and a unique reservation ID.

Output: Users receive a reservation confirmation message on the screen, indicating a successful reservation with all relevant details.

Test Cases: TBD

Requirement: 7

Use Case: 7: Alternative Transportation Information

Name: Alternative Transportation Information

Introduction: The system shall provide users with information about alternative transportation options to and from the parking location.

Rationale: To enhance user convenience and encourage the use of alternative transportation methods, such as public transit or carpooling.

Inputs: User's request for alternative transportation information.

Requirements Description: The system shall access and display information regarding public transportation routes, schedules, nearby carpooling options, and any special discounts or promotions related to alternative transportation.

Outputs: The system shall present alternative transportation information to the user, including details about public transportation routes, nearby carpooling options, and any available discounts or promotions.

Requirement: 8

Use Case: 8: User Feedback and Support

Name: User Feedback and Support

Introduction: The system shall provide a user feedback mechanism and offer customer support options to address user inquiries and issues.

Rationale: To ensure a positive user experience and resolve any problems or questions users may have.

Inputs: User-initiated requests for assistance or feedback submission.

Requirements Description: The system shall include a user-friendly feedback form where users can submit comments, questions, or issues. The system shall provide access to a customer support portal or chatbot for real-time assistance. The system shall log and categorize user feedback and inquiries for analysis and response.

Outputs: Users shall have the ability to submit feedback or inquiries. Users shall have access to customer support options, including a chatbot or support portal.

Requirement: 9

Use Case: 9

Name: Admin Parking Management

Introduction: Allows an authorized admin user to make changes to the parking system. The primary purpose is to provide administrators with the necessary tools to modify existing parking configurations or create new ones.

Rationale: The parking system remains flexible and responsive to changes. Admins need the capabilities to manage parking resources effectively.

Inputs: Admin user selects action to either modify existing parking or create new parking. Admin inputs mandatory data, which includes location detail, parking lot information (e.g., name, size, type), available parking spots, and any other relevant information.

Requirements Description: The system provides admin users the option to modify/ create parking configurations. When modifying, it allows admin to update relevant information. When creating, it allows admin to successfully input and create a parking system.

Outputs: Modified/new parking and notification to student users regarding changes made via emails they provided and within the application.

Test Cases:

Requirement: 10

Use Case: 10

Name: Admin Permit Management

Introduction: Allows an authorized admin user to make changes to permits within the parking system.

Rationale: The permit system remains flexible and responsive to changes. Admins need the capabilities to manage permit resources effectively.

Inputs: Admin user selects action to either modify existing permit or create new permit. Admin inputs mandatory data, which includes location detail, parking lot information (e.g., name, size, type), available parking spots, dates/times of availability, length of permit, and any other relevant information.

Requirements Description: The system provides admin users the option to modify/ create permit configurations. When modifying, it allows admin to update relevant information. When creating, it allows admin to successfully input and create a permit selection.

Outputs: Modified/new permits and notification to student users regarding changes made via emails they provided and within the application.

Test Cases:

Requirement: 11

Use Case: 11

Name: Admin User Feedback Analysis

Introduction:

Rationale:

Inputs:

Requirement Description:

Output:

Test Cases:

Requirement: 12

Use Case: 12

Name: Security and Privacy Management

Introduction:

Rationale:

Inputs:

Requirement Description:

Output:

Test Cases:

**SECTION 5:**

**Park-Me**

**Panther\_id(PC), Student\_first\_name, Student\_last\_name,**   
**Student\_phone\_number(FN), Student\_email, Membership\_Acess, Validity\_number**

**Validity\_number, Permission\_Acess, Park\_lots**

**Membership\_Acess, Validity\_number, Price\_charge**

**Panther\_id(PC), Start\_date, Membership\_Access, Validity\_number, Next\_charge\_date**

**Park\_lots, Location, Time, Open\_spots, Taken\_spots**

**Database Specifications:**

Database Mangement System:

**Use Case: 4**

Parent Table: Customer

Name of the table: Customer

* Customer\_id (Primary Key)
* Customer\_first\_name
* Customer\_last\_name
* Customer\_phone\_number (Foreign Key referencing Phone.Phone\_id)

Child Tables:

Name of the table: Phone

* Phone\_id (Primary Key)
* Phone\_number
* Name of the table: Permit
* Permit\_id (Primary Key)
* Customer\_id (Foreign Key referencing Customer.Customer\_id)
* Permit\_type
* Permit\_start\_date
* Permit\_end\_date
* Permit\_status
* Permit\_price

**Use Case: 5**

Parent Table: User

Name of the table: User

* User\_id (Primary Key)
* Username
* Email
* Password (hashed for security)

Child Tables:

Name of the table: CampusMap

* CampusMap\_id (Primary Key)
* Map\_name
* Map\_description
* Name of the table: MapLocation
* MapLocation\_id (Primary Key)
* CampusMap\_id (Foreign Key referencing CampusMap.CampusMap\_id)
* Location\_name
* Latitude
* Longitude
* Location\_description

Name of the table: UserLocation

* UserLocation\_id (Primary Key)
* User\_id (Foreign Key referencing User.User\_id)
* CampusMap\_id (Foreign Key referencing CampusMap.CampusMap\_id)
* Location\_id (Foreign Key referencing MapLocation.MapLocation\_id)
* Timestamp

**Use Case: 6**

Parent Tables:

* Name of the table: User
* User\_id (Primary Key)
* Username
* Email
* Password (hashed for security)

Name of the table: ParkingLot

* Lot\_id (Primary Key)
* Lot\_name
* Total\_capacity
* Current\_occupancy
* Location

Child Tables:

Name of the table: Reservation

* Reservation\_id (Primary Key)
* User\_id (Foreign Key referencing User.User\_id)
* Lot\_id (Foreign Key referencing ParkingLot.Lot\_id)
* Start\_time
* End\_time
* Reservation\_status
* Payment\_id (Foreign Key referencing Payment.Payment\_id)

Name of the table: Payment

* Payment\_id (Primary Key)
* User\_id (Foreign Key referencing User.User\_id)
* Amount
* Payment\_date
* Payment\_status
* Payment\_method

**Use Case 9: Admin Parking Management**

**Parent Tables:**

1. Admin (Admin\_ID, Admin\_FirstName, Admin\_LastName, Admin\_PhoneNumber, Admin\_Email, Admin\_PermissionLevel)
2. ParkingLots (ParkingLot\_id, ParkingLot\_Name, ParkingLot\_Location, ParkingLot\_Capacity)

**Use Case 10: Admin Permit Management**

**Parent Tables:**

1. Admin (Admin\_ID, Admin\_FirstName, Admin\_LastName, Admin\_PhoneNumber, Admin\_Email, Admin\_PermissionLevel)
2. Permits (Permit\_ID, Permits\_Name, Permits\_Description, Permits\_Price, Permits\_ValidatyPeriod)

**Child Tables:**

1. PermitAssignments (Assignment\_ID, Admin\_ID, Permit\_ID, Assigned\_Date, Assined\_EndDate)

**SECTION 6:**

[Github]