1. **User Account Registration**
   1. User must register with an email not already in the database
   2. User must register with a username not already in the database
   3. User must register with a password of at least 8 characters long
   4. System validates that all fields are entered correctly
   5. System validates that there are no conflicts with the database
      1. System shows an invalid prompt if there are conflicts
      2. System requests user to address conflicts
   6. Account System must record new account in database
2. **User Account Login**
   1. User must login with username and password
   2. System validates that all fields are entered correctly
   3. System validates that information entered matches with database
   4. System redirects user to the main page after successful validation
   5. If username entered is not in database, system will inform user that the username inputted was incorrect
   6. If username entered is in database but password entered is wrong, System will inform User of error.
      1. User is allowed a total of five tries to input the correct password
3. **Account Recovery**
   1. System sends an email verification associated with that username after the fifth failed attempt to reset the User’s password
      1. User is not allowed to attempt login to account until email verification is replied to
      2. E-mail verification link expires 5 minutes after e-mail was sent
   2. User is able to reset their password by clicking on “Forgot your password?” button on login screen
      1. System sends an email to the user to allow them to reset their password
      2. E-mail verification link expires 5 minutes after e-mail was sent
   3. Account System changes User’s password in database
4. **Location Service**
   1. System must be able to gather the User’s location through their device’s location function
   2. System must allow User to input their location manually
5. **Main Page**
   1. System must display a map, centred on the location the User has entered/provided by the location service
      1. Map shows area around User in 3 kilometre radius
   2. User must be able to scroll through the map
   3. User must be able to zoom in and out of the map
   4. User must be able to see their previous reviews
   5. User must be able to navigate to settings
6. **View Reviews**
   1. System must display all reviews made by User
   2. System must allow User to edit review made by User
   3. System must allow User to delete review made by User
7. **Review Making**
   1. User must be able to write reviews for restaurants
      1. Reviews are limited to 300 characters long
   2. User must be able to add ratings to restaurants
      1. Ratings are on a scale from 1 to 5
   3. User must be able to edit reviews
   4. User must be able to edit ratings
8. **Settings Menu**
   1. User must enter their password to enter the settings menu
      1. System will inform User if password inputted is incorrect
   2. User must be able to change their password
      1. User must input their current password correctly to change their password
         1. New password must be at least 8 characters long
         2. System will inform the user if password inputted does not meet requirements
      2. System will send an email to User’s email, informing them of their password change
   3. User must be able to change their email
      1. System will send an email to User’s new email to confirm the change
9. **View Nearby Restaurants**
   1. System must be able to retrieve restaurant information from database provided
   2. System must be able to display restaurants in a 3 kilometre radius from the location specified by User
      1. The displayed restaurants must show their cuisine type, proximity and ratings
   3. User must be able to input a restaurants name manually to view its information
   4. The restaurants should be displayed on the map overlay
   5. System must be able to filter the restaurants to display
      1. The filters are “cuisine” , “proximity” or “rating”
   6. System must be able to display a random restaurant in a 3 kilometre radius if the User selects the “random” button
      1. The random button takes in to account any filters applied on the System by the User
10. **View Specific Restaurants**
    1. User must be able to select a restaurant to view more information
       1. Information such as cuisine type, distance, rating and opening hours must be displayed
    2. User must be able to read reviews left by other users
    3. User must be able to add reviews to the restaurant
    4. User must be able to add ratings to the restaurant
       1. Ratings are on a scale from 1 to 5
11. **View Nearby Parking**
    1. System must show available parking spaces that are at most 1 kilometre away from the restaurant
       1. System must have a filter to lower search radius up to 0.5 kilometres
    2. User must be able to select a specific parking space from those displayed
12. **View Specific Parking**
    1. System must display the number of available lots and total number of lots at parking space specified
13. **Pathfinding**
    1. User must select “Get directions” at either the “Specific Restaurant” page or “Specific Parking Space” page
       1. System must redirect User to Google Maps, inputting the location for the User
14. **Non-functional Requirements**
    1. The system must be able to support up to 100 concurrent users
    2. The system must take up to 3 seconds to load under normal network conditions
    3. The system must be supported by the common web-browsers used
    4. The system must scale according to the screen size of the device used by the User
    5. The database must support standard SQL/MySQL queries

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: |  | | |
| Use Case Name: | User Account Registration | | |
| Created By: | Holdon | Last Updated By: |  |
| Date Created: | 05/02/2023 | Date Last Updated: |  |

|  |  |
| --- | --- |
| Actor: | User, Account System |
| Description: | Users are required to create an account to use the application |
| Preconditions: | User starts up the application |
| Postconditions: | User completes account registration |
| Priority: |  |
| Frequency of Use: |  |
| Flow of Events: | 1. User clicks on “Create a New Account” 2. System requests User to input e-mail, username and password 3. System checks if e-mail is already in database 4. System checks if username is already in database 5. System checks if password is at least 8 characters long 6. User is redirected to login page 7. Account System logs new account into database |
| Alternative Flows: | AF-3) : User enters an e-mail that is already in database   1. System informs User that e-mail has already been registered 2. System returns to 2)   AF-4) : User enters a username that is already in database   1. System informs User that username inputted has already been used 2. System returns to 2)   AF-5) : User enters a password that is not 8 characters long   1. System informs User that password inputted has to be at least 8 characters long 2. System returns to 2) |
| Exceptions: |  |
| Includes: |  |
| Special Requirements: |  |
| Assumptions: |  |
| Notes and Issues: |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: |  | | |
| Use Case Name: | User Account Login | | |
| Created By: | Holdon | Last Updated By: |  |
| Date Created: | 05/02/2023 | Date Last Updated: |  |

|  |  |
| --- | --- |
| Actor: | User, Account System, Email System |
| Description: | Users have to login with a registered account to access the application |
| Preconditions: | User has a registered account |
| Postconditions: | User successfully logs in |
| Priority: |  |
| Frequency of Use: |  |
| Flow of Events: | 1. User enters their username and password into the fields 2. Account System validates login credentials 3. System redirects User to main page |
| Alternative Flows: | AF 2) : User inputs invalid credentials   1. If User inputs a username not in the database, System will inform User that username is not in database 2. If User inputs a username in the database but inputs the wrong password, System will inform User of error 3. System returns to 1) unless User has failed to login 5 times 4. User is not allowed to login to account 5. Email System sends User an email to the email associated to that username for email verification |
| Exceptions: |  |
| Includes: |  |
| Special Requirements: |  |
| Assumptions: |  |
| Notes and Issues: |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: |  | | |
| Use Case Name: | Account Recovery | | |
| Created By: | Holdon | Last Updated By: |  |
| Date Created: | 05/02/2023 | Date Last Updated: |  |

|  |  |
| --- | --- |
| Actor: | User, Email System, Account System |
| Description: | User has ability to recover account if they forget their password |
| Preconditions: | User requests password reset  User has attempted 5 logins to 1 account unsuccessfully |
| Postconditions: | User recovers password |
| Priority: |  |
| Frequency of Use: |  |
| Flow of Events: | 1. Email System sends e-mail to User’s e-mail account after fifth unsuccessful login/requests password reset 2. User clicks on link on e-mail to reset their password 3. User enters new password 4. Account System updates database with new password |
| Alternative Flows: | AF 2) : User does not click on email link   1. If user does not click on link 5 minutes after e-mail was sent, the link will expire 2. No changes will be made |
| Exceptions: |  |
| Includes: |  |
| Special Requirements: |  |
| Assumptions: |  |
| Notes and Issues: |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: |  | | |
| Use Case Name: | Location Service | | |
| Created By: | Holdon | Last Updated By: |  |
| Date Created: | 05/02/2023 | Date Last Updated: |  |

|  |  |
| --- | --- |
| Actor: | User, Device Location |
| Description: | User must allow System to get their device location for the application to function |
| Preconditions: | User has logged in to System |
| Postconditions: | User returns location to System |
| Priority: |  |
| Frequency of Use: |  |
| Flow of Events: | 1. User enables their device’s location services 2. System records User’s current location |
| Alternative Flows: | AF 1) : If user has not enabled/does not have device location   1. User manually inputs their location 2. System returns to 2) |
| Exceptions: |  |
| Includes: |  |
| Special Requirements: |  |
| Assumptions: |  |
| Notes and Issues: |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: |  | | |
| Use Case Name: | Main Page | | |
| Created By: | Holdon | Last Updated By: |  |
| Date Created: | 05/02/2023 | Date Last Updated: |  |

|  |  |
| --- | --- |
| Actor: | User, Map System, Location System |
| Description: | Main page is the first thing the User sees after successful login. It contains the map where the User will navigate. |
| Preconditions: | User has successfully logged in. |
| Postconditions: | User leaves Main page |
| Priority: |  |
| Frequency of Use: |  |
| Flow of Events: | 1. Location System requests User to input location 2. Map System shows map centred at User’s location, showing area in 3 kilometre radius 3. User freely navigates map |
| Alternative Flows: |  |
| Exceptions: |  |
| Includes: |  |
| Special Requirements: |  |
| Assumptions: |  |
| Notes and Issues: |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: |  | | |
| Use Case Name: | View Reviews | | |
| Created By: | Holdon | Last Updated By: |  |
| Date Created: | 05/02/2023 | Date Last Updated: |  |

|  |  |
| --- | --- |
| Actor: | User, Reviews System |
| Description: | Reviews system contains all the reviews left by the User for reference |
| Preconditions: | User is on the main page |
| Postconditions: | User leaves “Reviews” page |
| Priority: |  |
| Frequency of Use: |  |
| Flow of Events: | 1. User selects “Reviews” on the main page 2. Reviews System displays reviews previously made by the User |
| Alternative Flows: |  |
| Exceptions: |  |
| Includes: | Review Making |
| Special Requirements: |  |
| Assumptions: |  |
| Notes and Issues: |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: |  | | |
| Use Case Name: | Review Making | | |
| Created By: | Holdon | Last Updated By: |  |
| Date Created: | 05/02/2023 | Date Last Updated: |  |

|  |  |
| --- | --- |
| Actor: | User, Reviews System |
| Description: | Reviews left by Users can be edited or deleted if the User chooses |
| Preconditions: | User is on View Reviews |
| Postconditions: | User makes changes to existing review |
| Priority: |  |
| Frequency of Use: |  |
| Flow of Events: | 1. User selects specific review on “View Reviews” page 2. User edits to review previously made 3. User edits rating to rating previously made |
| Alternative Flows: | AF 2) : User wants to delete rating   1. User can delete review previously made 2. System returns to “View Reviews” page |
| Exceptions: |  |
| Includes: |  |
| Special Requirements: |  |
| Assumptions: |  |
| Notes and Issues: |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: |  | | |
| Use Case Name: | Settings Menu | | |
| Created By: |  | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

|  |  |
| --- | --- |
| Actor: | User, Account System, E-mail System |
| Description: | User must be able to change their password or e-mail if they wish to do so |
| Preconditions: | User is on main page |
| Postconditions: | User exits settings menu |
| Priority: |  |
| Frequency of Use: |  |
| Flow of Events: | 1. User selects “Settings” on main page 2. User must enter their password 3. Account System verifies if password entered is correct 4. System displays settings menu 5. Settings menu has “Change Password” and “Change E-mail” |
| Alternative Flows: | AF 3) : User enters incorrect password   1. System informs User that password entered was incorrect 2. System returns to 2)   AF 5) : User wants to change password   1. User selects “Change Password” 2. User inputs current password 3. User inputs new password, which is 8 characters or longer 4. E-mail System sends e-mail to User to inform them of password change   AF 5) : User wants to change e-mail   1. User selects “Change E-mail” 2. User inputs new E-mail 3. E-mail System sends e-mail to User’s new e-mail to confirm the change |
| Exceptions: |  |
| Includes: |  |
| Special Requirements: |  |
| Assumptions: |  |
| Notes and Issues: |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: |  | | |
| Use Case Name: | View Nearby Restaurants | | |
| Created By: |  | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

|  |  |
| --- | --- |
| Actor: |  |
| Description: |  |
| Preconditions: |  |
| Postconditions: |  |
| Priority: |  |
| Frequency of Use: |  |
| Flow of Events: |  |
| Alternative Flows: |  |
| Exceptions: |  |
| Includes: |  |
| Special Requirements: |  |
| Assumptions: |  |
| Notes and Issues: |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: |  | | |
| Use Case Name: | View Specific Restaurant | | |
| Created By: |  | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

|  |  |
| --- | --- |
| Actor: |  |
| Description: |  |
| Preconditions: |  |
| Postconditions: |  |
| Priority: |  |
| Frequency of Use: |  |
| Flow of Events: |  |
| Alternative Flows: |  |
| Exceptions: |  |
| Includes: |  |
| Special Requirements: |  |
| Assumptions: |  |
| Notes and Issues: |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: |  | | |
| Use Case Name: | View Nearby Parking | | |
| Created By: |  | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

|  |  |
| --- | --- |
| Actor: |  |
| Description: |  |
| Preconditions: |  |
| Postconditions: |  |
| Priority: |  |
| Frequency of Use: |  |
| Flow of Events: |  |
| Alternative Flows: |  |
| Exceptions: |  |
| Includes: |  |
| Special Requirements: |  |
| Assumptions: |  |
| Notes and Issues: |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: |  | | |
| Use Case Name: | View Specific Parking | | |
| Created By: |  | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

|  |  |
| --- | --- |
| Actor: |  |
| Description: |  |
| Preconditions: |  |
| Postconditions: |  |
| Priority: |  |
| Frequency of Use: |  |
| Flow of Events: |  |
| Alternative Flows: |  |
| Exceptions: |  |
| Includes: |  |
| Special Requirements: |  |
| Assumptions: |  |
| Notes and Issues: |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: |  | | |
| Use Case Name: | Pathfinding | | |
| Created By: |  | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

|  |  |
| --- | --- |
| Actor: |  |
| Description: |  |
| Preconditions: |  |
| Postconditions: |  |
| Priority: |  |
| Frequency of Use: |  |
| Flow of Events: |  |
| Alternative Flows: |  |
| Exceptions: |  |
| Includes: |  |
| Special Requirements: |  |
| Assumptions: |  |
| Notes and Issues: |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: |  | | |
| Use Case Name: |  | | |
| Created By: |  | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

|  |  |
| --- | --- |
| Actor: |  |
| Description: |  |
| Preconditions: |  |
| Postconditions: |  |
| Priority: |  |
| Frequency of Use: |  |
| Flow of Events: |  |
| Alternative Flows: |  |
| Exceptions: |  |
| Includes: |  |
| Special Requirements: |  |
| Assumptions: |  |
| Notes and Issues: |  |