**Use Cases**

Hangout

**Version 1.0 approved**

**Prepared by Jethro Phuah, Zhi Fah, Ernest Ang, Alicia Chua, Shannon, Sim Hao, Shaun Ong**

**Team Mac & Cheese**

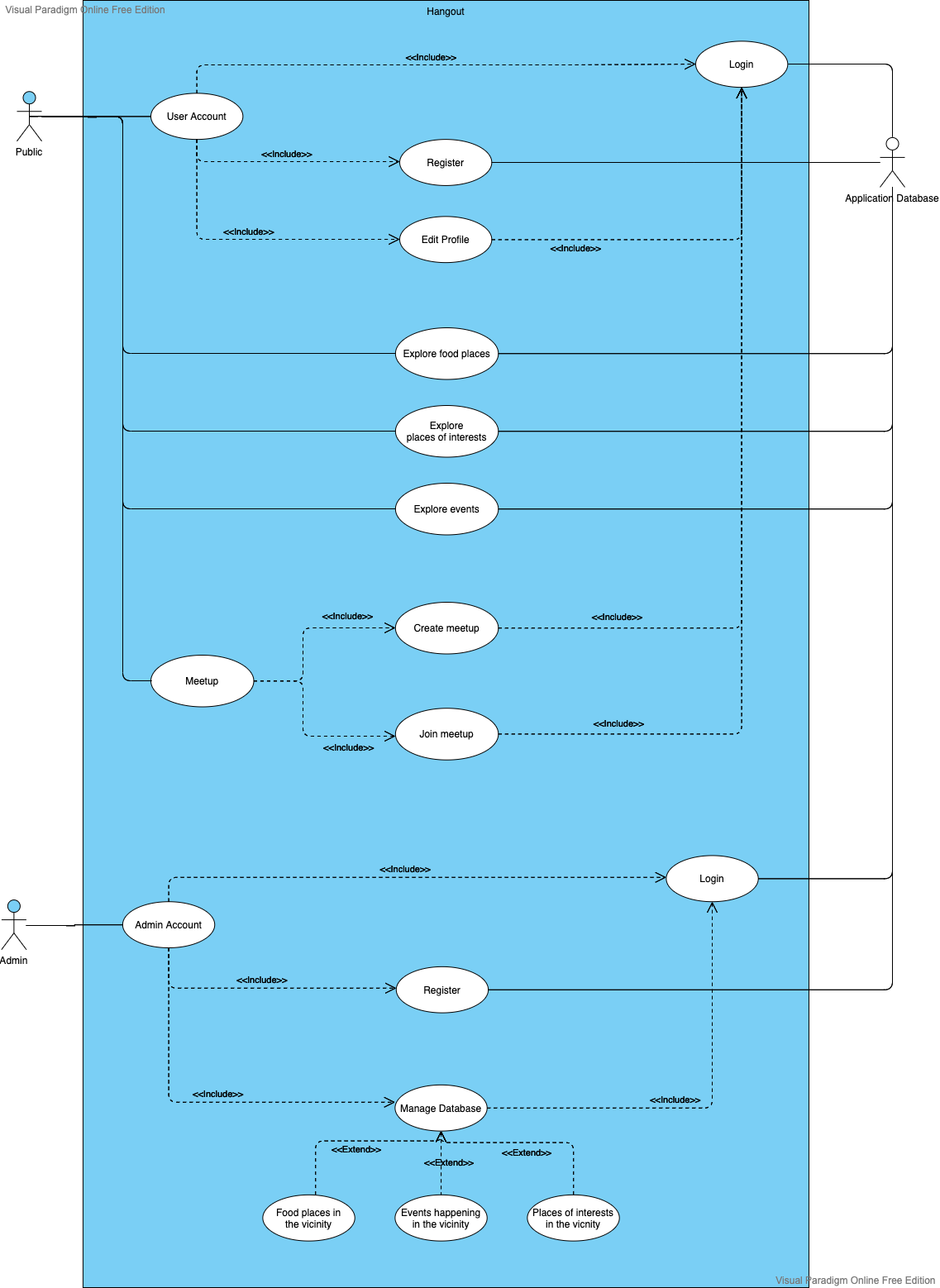
**22 August 2021**

**Revision History**

| **Name** | **Date** | **Reason For Changes** | **Version** |
| --- | --- | --- | --- |
| Jethro Phuah | 22/8/21 | Added use case descriptions | 1.0 |
| Lam Zhi Fah | 24/8/21 | Added use case descriptions | 1.0 |
| Ernest Ang | 25/8/21 | Added use case diagrams, use case descriptions | 1.0 |

**Use Case List**

| ***Primary Actor*** | ***Use Cases*** |
| --- | --- |
| Public | Create account |
| Public | Log in to system via registered email address |
| Public | Edit profile to add favorite food preferences and locations of interest |
| Public | Choose based on current (GPS) or selected location |
| Public | Find food options. |
| Public | Find places of interest that are either outdoor or indoor |
| Public | Events happening around them |
| Public | Create Meetups |
| Public | Join Meetups |
| Admin | Create admin account |
| Admin | Log in to system via registered email address |
| Admin | Add places of interest for the GRC admin belongs to (e.g. Tampines) |
| Admin | Add food options for the GRC admin belongs to (e.g. Tampines) |
| Admin | Add event happenings for the GRC admin belongs to (e.g. Tampines) |

**Use Case Template**

| Use Case ID: | UC001 | | |
| --- | --- | --- | --- |
| Use Case Name: | Create account | | |
| Created By: |  | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

| Actors: | Public |
| --- | --- |
| Description: | The user can create his/her account. |
| Trigger: |  |
| Preconditions: | The user does not have an existing account. |
| Postconditions: | The user has created his/her account. |
| Normal Flow: | 1. The user clicks on ‘Sign up’. 2. The user indicates his/her preferred username, email address, handphone number and password. 3. The user can choose to agree to receive newsletters and notifications. 4. The user will be required to agree to Terms and Conditions. 5. The user will be required to re-enter password as confirmation. 6. The user clicks on ‘Register’ to finish creating the account. 7. A message “Account created successfully” is displayed on the screen. |
| Alternative Flows: | AF-S6-1: If the user does not accept Terms and Conditions:   1. The system displays “Please agree to Terms and Conditions”. 2. The system returns to step 4.   AF-S6-2: If the email address is already in use:   1. The system displays “Email already in use”. 2. The system returns to step 2.   AF-S6-3: If the password is lesser than 6 characters:   1. The system displays “Weak password”. 2. The system returns to step 2.   AF-S6-4: If the passwords do not match:   1. The system displays “Password does not match”. 2. The system returns to step 2.   AF-S6-5: If any of the input is missing:   1. The system displays “Missing input”, depending on which field is missing. 2. The system returns to step 2. |
| Exceptions: | - |
| Includes: | - |
| Priority: | High |
| Frequency of Use: | Low |
| Business Rules: | - |
| Special Requirements: | - |
| Assumptions: | - |
| Notes and Issues: | - |

| Use Case ID: | UC002 | | |
| --- | --- | --- | --- |
| Use Case Name: | Log In | | |
| Created By: |  | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

| Actors: | Public |
| --- | --- |
| Description: | The user logs in to the game system via a registered email address. |
| Trigger: |  |
| Preconditions: | The user has successfully started up the application. |
| Postconditions: | The user has logged in to the app with an account. |
| Normal Flow: | 1. The user opens the application. 2. The system displays the login page. 3. The user enters his/her username and password. 4. The user clicks the login button 5. The user successfully logs in to the system. 6. A message “Log in Successfully” is displayed on the screen. |
| Alternative Flows: | AF-S4-1: If the authentication fails:   1. The system displays “Incorrect password” or “Invalid email”, depending on which field is invalid. 2. The system returns to Step 2.   AF-S4-2: If the email or password field is missing:   1. The system displays “Missing email” or “Missing Password” respectively. 2. The system returns to Step 2.   AF-S4-3: If the user account does not exist:   1. The system displays “Account does not exist”. 2. The system returns to Step 2. 3. The user will have to register for an account. |
| AF-S3: If they forgot password:   1. The user clicks on “Forgot Password”. 2. The system displays the email verification page. 3. The user enters their NTU email address into the text field. 4. The system checks the email validity, and displays email sent. 5. The user will go to their inbox to reset his/her password. |
| Exceptions: | - |
| Includes: | - |
| Priority: | High |
| Frequency of Use: | Low |
| Business Rules: | - |
| Special Requirements: | - |
| Assumptions: | - |
| Notes and Issues: | - |

| Use Case ID: | UC003 | | |
| --- | --- | --- | --- |
| Use Case Name: | Edit Profile | | |
| Created By: |  | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

| Actors: | Public |
| --- | --- |
| Description: | The user can edit his/her account. |
| Trigger: |  |
| Preconditions: | The user has an existing account. |
| Postconditions: | The user has updated his profile settings. |
| Normal Flow: | 1. The user clicks on ‘Edit Profile’. 2. The user can indicate dietary preferences and types of activities. 3. The user can add highly-frequented activity locations and dining places under ‘Favourites’. 4. The user can edit notification preferences. |
| Alternative Flows: | ? |
| Exceptions: | - |
| Includes: | - |
| Priority: | High |
| Frequency of Use: | Medium |
| Business Rules: | - |
| Special Requirements: | - |
| Assumptions: | - |
| Notes and Issues: | - |

| Use Case ID: | UC004 | | |
| --- | --- | --- | --- |
| Use Case Name: | Select Location | | |
| Created By: |  | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

| Actors: | Public |
| --- | --- |
| Description: | The user can select his location to be either his/her current GPS location or location of choice. |
| Trigger: |  |
| Preconditions: | The user has successfully logged in to the application. |
| Postconditions: | The user is directed to the home page with his current GPS location as default. |
| Normal Flow: | 1. The user can scroll through the 3 main categories “Food”, “Places of interest” and “Events” under the Home Page with his/her GPS location. 2. The user can choose to change his location to his/her preferred location. |
| Alternative Flows: |  |
| Exceptions: | - |
| Includes: | - |
| Priority: | High |
| Frequency of Use: | Medium |
| Business Rules: | - |
| Special Requirements: | - |
| Assumptions: | - |
| Notes and Issues: | - |

| Use Case ID: | UC005 | | |
| --- | --- | --- | --- |
| Use Case Name: | Find Food | | |
| Created By: |  | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

| Actors: | Public |
| --- | --- |
| Description: | The user can explore food of his choice. |
| Trigger: |  |
| Preconditions: | The user selects the “Food” option. |
| Postconditions: | - |
| Normal Flow: | 1. The user clicks on the “Food” option. 2. The user selects the cuisine of his/her choice. 3. The user can filter the stalls according to price, dining type and its different cuisines. 4. The user can search for his/her preferred stall/restaurant. 5. The user selects the stall of his/her choice to view its address, available menu and opening days and hours. 6. The user can add the food stall to his/her favourites tab. |
| Alternative Flows: |  |
| Exceptions: |  |
| Includes: | - |
| Priority: | High |
| Frequency of Use: | High |
| Business Rules: | - |
| Special Requirements: | - |
| Assumptions: | - |
| Notes and Issues: | - |

| Use Case ID: | UC006 | | |
| --- | --- | --- | --- |
| Use Case Name: | Find places of interest | | |
| Created By: |  | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

| Actors: | Public |
| --- | --- |
| Description: | The user can explore places of interest around him/her. |
| Trigger: |  |
| Preconditions: | The user selects the “Places of Interest” option. |
| Postconditions: | - |
| Normal Flow: | 1. The user clicks on the “Places of Interest” option. 2. The user selects the category of his/her choice. 3. The user selects the place of his/her interest to view its address, and opening days and hours. 4. The user can add the palace of interest to his/her favourites tab. |
| Alternative Flows: |  |
| Exceptions: | - |
| Includes: | - |
| Priority: | High |
| Frequency of Use: | High |
| Business Rules: | - |
| Special Requirements: | - |
| Assumptions: | - |
| Notes and Issues: | - |

| Use Case ID: | UC007 | | |
| --- | --- | --- | --- |
| Use Case Name: | Events | | |
| Created By: |  | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

| Actors: | Public |
| --- | --- |
| Description: | The user can view the upcoming events happening in his/her chosen vicinity. |
| Trigger: |  |
| Preconditions: | The user selects “Events happening around here”. |
| Postconditions: | - |
| Normal Flow: | 1. The user clicks on ‘Events’. 2. The user can select from a list of upcoming events that are happening around the chosen vicinity. 3. The user can view the time, location and sign up fee (if needed) for each of the events. 4. The user can add the event of interest to his/her favourites tab. |
| Alternative Flows: |  |
| Exceptions: | - |
| Includes: | - |
| Priority: | High |
| Frequency of Use: | Medium |
| Business Rules: | - |
| Special Requirements: | - |
| Assumptions: | - |
| Notes and Issues: | - |

| Use Case ID: | UC008 | | |
| --- | --- | --- | --- |
| Use Case Name: | Create Meetup | | |
| Created By: |  | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

| Actors: | Public |
| --- | --- |
| Description: | The user can create meetup events. |
| Trigger: |  |
| Preconditions: | The user selects “Create Meetup”. |
| Postconditions: | - |
| Normal Flow: | 1. The user can create meetups on on the district chosen previously. 2. The user have to set a time for the meetup 3. The user have to set the address for the meet up. 4. The user have to write a short description for the meetup. 5. The user have to give a name to the meetup. 6. The user can include pictures relating to the meetup. 7. The user can accept or reject other user’s request to join the meetup. |
| Alternative Flows: |  |
| Exceptions: | - |
| Includes: | - |
| Priority: | High |
| Frequency of Use: | Medium |
| Business Rules: | - |
| Special Requirements: | - |
| Assumptions: | - |
| Notes and Issues: | - |

| Use Case ID: | UC009 | | |
| --- | --- | --- | --- |
| Use Case Name: | Join meetup | | |
| Created By: |  | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

| Actors: | Public |
| --- | --- |
| Description: | The user can join meetup events. |
| Trigger: |  |
| Preconditions: | The user selects “Meetups”. |
| Postconditions: | - |
| Normal Flow: | 1. The user clicks on ‘Meetups’. 2. The user can select from a list of upcoming meetups that are happening around the chosen vicinity. 3. The user can search meetups based on keywords. 4. The user can view the time, location, description and list of people who joined for each of the meetups. 5. The user can request to join which is subject to the decision of the creator. 6. The user receives a notification of entry if his/her request is accepted. 7. The user will be added into the list of people in the meetup. |
| Alternative Flows: | AF-S4-1: If the user’s request is denied:   1. The user receives a notification of entry if his/her request is denied. 2. The user can return to find other meetups. |
| Exceptions: | - |
| Includes: | - |
| Priority: | High |
| Frequency of Use: | Medium |
| Business Rules: | - |
| Special Requirements: | - |
| Assumptions: | - |
| Notes and Issues: | - |

| Use Case ID: | UC010 | | |
| --- | --- | --- | --- |
| Use Case Name: | Create Admin Account | | |
| Created By: |  | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

| Actors: | Admin |
| --- | --- |
| Description: | Admin can create an account for his/her respective GRC. |
| Preconditions | Account for a specific GRC must not have been already created . |
| Postconditions: | The user has created his/her account for that GRC. |
| Normal Flow: | 1. The user clicks on ‘Sign up’. 2. The user indicates his/her preferred username, email address, GRC, handphone number and password. 3. The user will be required to agree to Terms and Conditions. 4. The user will be required to re-enter password as confirmation. 5. The user clicks on ‘Register’ to finish creating the account. 6. A message “Account created successfully” is displayed on the screen. |
| Alternative Flows: | AF-S5-1: If the user does not accept Terms and Conditions:   1. The system displays “Please agree to Terms and Conditions”. 2. The system returns to step 3   AF-S5-2: If the email address is already in use:   1. The system displays “Email already in use”. 2. The system returns to step 2.   AF-S5-3: If the password is lesser than 6 characters:   1. The system displays “Weak password”. 2. The system returns to step 2.   AF-S5-4: If the passwords do not match:   1. The system displays “Password does not match”. 2. The system returns to step 2.   AF-S5-5: If any of the input is missing:   1. The system displays “Missing input”, depending on which field is missing. 2. The system returns to step 2.   AF-S5-6: If there is already an account for GRC input:   1. The system displays “Account already exist for GRC” 2. The system returns to step 2. |
| Exceptions: | - |
| Includes: | - |
| Priority: | High |
| Frequency of Use: | Low |
| Business Rules: | - |
| Special Requirements: | - |

| Use Case ID: | UC011 | | |
| --- | --- | --- | --- |
| Use Case Name: | Admin Login | | |
| Created By: |  | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

| Actors: | Admin |
| --- | --- |
| Description: | The user logs in to the game system via a registered email address. |
| Trigger: |  |
| Preconditions: | The user has successfully started up the application. |
| Postconditions: | The user has logged in to the app with an account. |
| Normal Flow: | 1. The user opens the application. 2. The system displays the login page. 3. The user enters his/her username, GRC and password. 4. The user clicks the login button 5. The user successfully logs in to the system. 6. A message “Log in Successfully” is displayed on the screen. |
| Alternative Flows: | AF-S4-1: If the authentication fails:   1. The system displays “Incorrect password” or “Invalid email”, depending on which field is invalid. 2. The system returns to Step 2.   AF-S4-2: If the email or password field is missing:   1. The system displays “Missing email” or “Missing Password” respectively. 2. The system returns to Step 2.   AF-S4-3: If the user account does not exist:   1. The system displays “Account does not exist”. 2. The system returns to Step 2. 3. The user will have to register for an account. |
| AF-S3: If they forgot password:   1. The user clicks on “Forgot Password”. 2. The system displays the email verification page. 3. The user enters their NTU email address into the text field. 4. The system checks the email validity, and displays email sent. 5. The user will go to their inbox to reset his/her password. |
| Exceptions: | - |
| Includes: | - |
| Priority: | High |
| Frequency of Use: | Low |
| Business Rules: | - |
| Special Requirements: | - |
| Assumptions: | - |
| Notes and Issues: | - |

| Use Case ID: | UC012 | | |
| --- | --- | --- | --- |
| Use Case Name: | Add places of interest for the GRC which admin belongs to (e.g. Tampines) | | |
| Created By: |  | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

| Actors: | Admin |
| --- | --- |
| Description: | The user will be allowed to add places of interest for the GRC which the admin belongs to. |
| Trigger: |  |
| Preconditions: | The user has successfully logged in to the application. |
| Postconditions: | The user will create a place of interest under the user’s GRC |
| Normal Flow: | 1. Admin will input the name, description, and geolocation coordinates of the place of interest 2. Admin may add pictures of the places of interest. 3. Admin will click “Create” and create information about the place of interest. 4. System will display a “Successful” message. |
| Alternative Flows: | AF-S3-1: If any of the necessary field are empty:   1. The system displays “Missing name” or “Missing description” or “Missing coordinates” 2. The system returns to Step 1. |
| Exceptions: | - |
| Includes: | - |
| Priority: | High |
| Frequency of Use: | Medium |
| Business Rules: | - |
| Special Requirements: | - |
| Assumptions: | - |
| Notes and Issues: | - |

| Use Case ID: | UC013 | | |
| --- | --- | --- | --- |
| Use Case Name: | Add food options for the GRC which admin belongs to (e.g. Tampines) | | |
| Created By: |  | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

| Actors: | Admin |
| --- | --- |
| Description: | The user will be allowed to add food options for the GRC which admin belongs to. |
| Trigger: |  |
| Preconditions: | The user has successfully logged in to the application. |
| Postconditions: | The user will create a food option under the user’s GRC. |
| Normal Flow: | 1. Admin will input the name, description, and geolocation coordinates of the food option. 2. Admin may add pictures of the food option. 3. Admin will click “Create” and create information about the food option. 4. System will display a “Successful” message. |
| Alternative Flows: | AF-S3-1: If any of the necessary field are empty:   1. The system displays “Missing name” or “Missing description” or “Missing coordinates” 2. The system returns to Step 1. |
| Exceptions: | - |
| Includes: | - |
| Priority: | High |
| Frequency of Use: | Medium |
| Business Rules: | - |
| Special Requirements: | - |
| Assumptions: | - |
| Notes and Issues: | - |

| Use Case ID: | UC014 | | |
| --- | --- | --- | --- |
| Use Case Name: | Add event happenings for the GRC which admin belongs to (e.g. Tampines) | | |
| Created By: |  | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

| Actors: | Admin |
| --- | --- |
| Description: | The user will be allowed to add events happening for the GRC which admin belongs to. |
| Trigger: |  |
| Preconditions: | The user has successfully logged in to the application. |
| Postconditions: | The user will create an event happening for the GRC |
| Normal Flow: | 1. Admin will input the name, description, datetime and geolocation coordinates of the food option. 2. Admin may add pictures of the food option. 3. Admin will click “Create” and create information about the event happening. 4. System will display a “Successful” message. |
| Alternative Flows: | AF-S3-1: If any of the necessary field are empty:   1. The system displays “Missing name” or “Missing description” or “Missing coordinates” or “Missing datetime”. 2. The system returns to Step 1. |
| Exceptions: | - |
| Includes: | - |
| Priority: | High |
| Frequency of Use: | Medium |
| Business Rules: | - |
| Special Requirements: | - |
| Assumptions: | - |
| Notes and Issues: | - |