**Use Case Descriptions**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | 1.1 | | |
| Use Case Name: | Check Dengue Cases in Singapore for That Day | | |
| Created By: | Xing Xiang | Last Updated By: | - |
| Date Created: | 05/09/20 | Date Last Updated: | - |

|  |  |
| --- | --- |
| Actor: | User, database |
| Description: | The country-wide number of dengue cases for the current day is shown to the User. |
| Preconditions: | 1. The app boots successfully 2. The user’s smartphone has internet connection |
| Postconditions: | 1. The app successfully shows the country-wide number of cases for the current day to the user |
| Priority: | High |
| Frequency of Use: | 1-2 times per day |
| Flow of Events: | 1. The user opens the app 2. The app loads and presents the user with an input bar to input a specific location 3. The user selects the location from a dropdown box 4. The app queries the current day’s island-wide case count from the database 5. The app presents the case count to the user |
| Alternative Flows: | AF-S4: If the database has not been updated from nea.gov.sg for the current day   1. The app uses the most recent set of data for step 5. 2. The app displays the message “This data is correct as of <insert date>” |
| Exceptions: |  |
| Includes: |  |
| Special Requirements: |  |
| Assumptions: |  |
| Notes and Issues: |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | 1.2 | | |
| Use Case Name: | Check Dengue Cases in a Specified Location | | |
| Created By: | Xing Xiang | Last Updated By: | Xing Xiang |
| Date Created: | 28/08/20 | Date Last Updated: | 12/09/20 |

|  |  |
| --- | --- |
| Actor: | User, Database |
| Description: | The user must select a specific location and is presented with a dashboard with the most recent 14-day total number of dengue cases, as well as the trend in the number of dengue cases for the past week. The interface will be color coded based on the derived dengue alert level. |
| Preconditions: | 1. The app boots successfully 2. The user chooses the option to select a location 3. The user’s smartphone has internet connection |
| Postconditions: | 1. The app successfully shows the current 14-day-total cases to the user 2. The app interface is coloured in the correct dengue alert colour code |
| Priority: | High |
| Frequency of Use: | 1-2 times per day |
| Flow of Events: | 1. The user opens the app 2. The app loads and presents the user with an input bar to input a specific location 3. The user selects the location from a dropdown box 4. The app receives the location and sends a request to the Database API for all 14-day-total dengue cases in Singapore 5. The app filters the number of dengue cases in the selected location 6. The app presents the current 14-day-total cases to the user |
| Alternative Flows: | AF-S4: If the database has not been updated from data.gov.sg API   1. The app uses the previous set of data for steps 4-6. 2. The app displays the message “This data is correct as of <insert date>” |
| Exceptions: | EX1: The app loses internet connection while querying the API   1. The app displays “Oh no! We seem to have lost connection” 2. The app retries every 2 seconds to query the API once internet connection is back |
| Includes: | 1.1: Check Dengue Cases in Singapore for That Day  1.3: Select Location  3.1: Show Trends  4.1: Derive Dengue Alert Colour Code |
| Special Requirements: |  |
| Assumptions: |  |
| Notes and Issues: |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | 1.3 | | |
| Use Case Name: | Select Location | | |
| Created By: | Xing Xiang | Last Updated By: | - |
| Date Created: | 28/8/20 | Date Last Updated: | - |

|  |  |
| --- | --- |
| Actor: | User |
| Description: | The user selects their location from a dropdown box. |
| Preconditions: | 1. The “Select Location” dropdown box loads successfully 2. Permissions to get the user’s current location has been granted |
| Postconditions: | 1. The location entered is saved and used for subsequent actions |
| Priority: | High |
| Frequency of Use: | 1-2 times per day |
| Flow of Events: | 1. The user taps the dropdown box 2. The app presents the user with a dropdown list of various locations, including a “Current Location” option at the top 3. The user selects a location |
| Alternative Flows: |  |
| Exceptions: |  |
| Includes: |  |
| Special Requirements: |  |
| Assumptions: |  |
| Notes and Issues: |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | 2.1 | | |
| Use Case Name: | Report Potential Breeding Spot | | |
| Created By: | Darren | Last Updated By: | Darren |
| Date Created: | 05/09/20 | Date Last Updated: | 12/09/20 |

|  |  |
| --- | --- |
| Actor: | User, Email System |
| Description: | The user is able to report an area that he/she thinks is a potential breeding spot. After information about this potential breeding spot is provided by the user, it is then sent out to an external email system, and an email will be sent to the NEA to report this potential breeding spot |
| Preconditions: | 1. The app boots successfully 2. The user selects the option to report potential breeding spot 3. The user’s smartphone has internet connection |
| Postconditions: | 1. The app successfully displays an acknowledgement message to the user |
| Priority: | High |
| Frequency of Use: | 1-2 times per day |
| Flow of Events: | 1. The user opens the app 2. The app loads and presents the user with 2 options: select a location, or report potential breeding spots 3. The user selects the option to report potential breeding spots 4. The user is then prompted by the system to fill up a form with information about the potential breeding site. Details about the form are provided in use case 2.2 5. The user is provided with a non-mandatory option to upload a photo of the potential breeding site 6. The app displays a message that acknowledges that the details have been successfully received, and an email will be sent to the NEA |
| Alternative Flows: | AF-S5: If the user chooses not the upload a photo of the potential breeding site   1. Step 5 is skipped |
| Exceptions: | EX1: The app loses internet connection while connecting to the external email system   1. The app displays “The information has been recorded, but an email is unable to be sent out now due to technical difficulties. The potential breeding spot will be reported to the NEA once these issues are resolved. ” 2. The app retries every 2 seconds to connect to the external email system once the internet connection is back |
| Includes: | 2.1: Fill up form |
| Special Requirements: |  |
| Assumptions: |  |
| Notes and Issues: |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | 2.2 | | |
| Use Case Name: | Fill Up Form | | |
| Created By: | Darren | Last Updated By: | Darren |
| Date Created: | 05/09/20 | Date Last Updated: | 12/09/20 |

|  |  |
| --- | --- |
| Actor: | User |
| Description: | The user is required to fill up a form with details about the potential breeding spot. All of the input fields are situated within a single-page form, with only some fields being mandatory |
| Preconditions: | 1. The app boots successfully 2. The user selects the option to report potential breeding spot 3. The user’s smartphone has internet connection |
| Postconditions: | 1. The app moves on to use case 2.3 |
| Priority: | High |
| Frequency of Use: | 1-2 times per day |
| Flow of Events: | 1. The user is requested to input the date and time of occurrence (required field) 2. The user inputs street name (required field) 3. The user inputs building name (optional field) 4. The user inputs block or house number (optional field) 5. The user inputs the floor number (optional field) 6. The user inputs the unit number (optional field) 7. The user inputs the postal code (optional field) 8. The user inputs a description of the potential breeding site, capped at 5000 characters (required field) 9. The user clicks on the ‘submit’ button 10. The app saves the information provided by the user |
| Alternative Flows: |  |
| Exceptions: | EX1: The user does not fill up all required fields   1. The app displays an error message “Please fill up all required fields before proceeding” 2. The app displays the same form again, with the user’s previous inputs already filled |
| Includes: |  |
| Special Requirements: |  |
| Assumptions: |  |
| Notes and Issues: |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | 2.3 | | |
| Use Case Name: | Upload Image | | |
| Created By: | Darren | Last Updated By: |  |
| Date Created: | 05/09/20 | Date Last Updated: |  |

|  |  |
| --- | --- |
| Actor: | User |
| Description: | The user is able to upload an image of the potential breeding spot |
| Preconditions: | 1. The app boots successfully 2. The user selects the option to report potential breeding spot 3. The user has filled up the form (Use case 2.2 has been executed successfully) 4. The user’s smartphone has internet connection |
| Postconditions: | 1. Control is passed back to Use case 2.1 |
| Priority: | High |
| Frequency of Use: | 1-2 times per day |
| Flow of Events: | 1. The user is presented with an option to upload an image or to take a photo 2. The app asks for the user’s permission to access his/her camera 3. The user takes a photo of the potential breeding site 4. The user clicks on the ‘upload’ button 5. The app displays an acknowledgement message: “Your photo has been successfully uploaded.” |
| Alternative Flows: | AF-S2: If the user chooses to upload a photo already in their library instead of taking a photo   1. The app asks for the user’s permission to access his/her photo library 2. The user chooses a photo to upload |
| Exceptions: | EX1: The size of the image uploaded exceeds 3MB   1. The app displays “The size of the uploaded image is too large. Please upload another image with file size less than 3MB.” 2. Steps 1-5 are repeated |
| Includes: |  |
| Special Requirements: |  |
| Assumptions: |  |
| Notes and Issues: |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | 3.1 | | |
| Use Case Name: | Show Trends | | |
| Created By: | Dylan | Last Updated By: | - |
| Date Created: | 05/09/20 | Date Last Updated: | - |

|  |  |
| --- | --- |
| Actor: | Database |
| Description: | The user is able to observe trends of dengue cases of the past 14 days from a bar chart |
| Preconditions: | 1. The app boots successfully 2. The user has selected a specific location 3. The user’s smartphone has internet connection |
| Postconditions: | 1. A bar chart detailing the number of dengue cases each day for the past 14 days will be displayed |
| Priority: | High |
| Frequency of Use: | 1-2 times per day |
| Flow of Events: | 1. The user is presented with the option to view trends at the specified input location 2. The app queries the database and retrieves the stored data of dengue cases in the past 14 days 3. The app displays the data in a bar chart form on the interface |
| Alternative Flows: | AF-S4: If the database has not been updated from data.gov.sg API   1. The app uses the previous set of data for steps 2-3. 2. The app displays the message “This data is correct as of <insert date>” |
| Exceptions: | EX1: The specified location does not have a data set of the past 14 days   1. The app displays “Dengue case trend not available for this location. Please input a nearby location.” 2. Steps 1-3 are repeated |
| Includes: |  |
| Special Requirements: |  |
| Assumptions: |  |
| Notes and Issues: |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | 4.1 | | |
| Use Case Name: | Derive Dengue Alert Colour Code | | |
| Created By: | Dylan | Last Updated By: |  |
| Date Created: | 05/09/20 | Date Last Updated: |  |

|  |  |
| --- | --- |
| Actor: | Database |
| Description: | The app will determine the respective dengue alert colour code based on the number of dengue cases in the location. The display interface will imitate the colour of the dengue alert colour code. |
| Preconditions: | 1. The app boots successfully 2. The user has selected a specific location 3. The user’s smartphone has internet connection |
| Postconditions: | 1. The interface display colour changes to the dengue alert colour |
| Priority: | High |
| Frequency of Use: | 1-2 times per day |
| Flow of Events: | 1. The user is presented with an option to input a specific location 2. The app queries database for the dengue alert colour code of the specified location 3. The app changes the colour of its interface to match the derived colour from the database |
| Alternative Flows: |  |
| Exceptions: |  |
| Includes: |  |
| Special Requirements: |  |
| Assumptions: |  |
| Notes and Issues: |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | 5.1 | | |
| Use Case Name: | View Dengue Prevention Resources | | |
| Created By: | Wanyi | Last Updated By: |  |
| Date Created: | 08/09/20 | Date Last Updated: |  |

|  |  |
| --- | --- |
| Actor: | The user is able to select from 4 different choices on the app interface to view different information regarding dengue prevention, recognizing dengue symptoms, and a hotline connecting them to NEA. |
| Description: | The app will display the information depending on the choice selected by the user. These information have been added into the app beforehand as text or graphics. |
| Preconditions: | 1. The app boots successfully 2. The user has selected a choice of resource to view |
| Postconditions: | 1. The interface displays text, graphics or flowcharts according to the choice made by the user |
| Priority: | Medium |
| Frequency of Use: | 1-2 times per day |
| Flow of Events: | 1. The user is presented with various options to select from 2. The app displays the information that we have input for each selection |
| Alternative Flows: |  |
| Exceptions: |  |
| Includes: |  |
| Special Requirements: |  |
| Assumptions: |  |
| Notes and Issues: |  |