

Le Fleur King

Elaboration Document

Rosy Outlook

Team Members:

|  |
| --- |
| 224078321, Ganeefa Sirkhotte  223126756, Junaid Brooks |

Document Date: 28 May 2023

**TABLE OF CONTENTS**

[1 FUNCTIONAL REQUIREMENTS 3](#_Toc136184563)

[1.1 Analysis Use Case Model 3](#_Toc136184564)

[1.2 Use Case Glossary and Responsibilities 4](#_Toc136184565)

[2 UI Prototypes 6](#_Toc136184566)

[2.1 Team UI Guidelines 6](#_Toc136184567)

[2.2 Team Design for 8](#_Toc136184568)

[2.2.1 Login / Logout 8](#_Toc136184569)

[2.2.2 Dashboard / Landing Page / Main Page 9](#_Toc136184570)

[2.3 UI Designs & Updated Analysis Use Case Narratives 12](#_Toc136184571)

[2.3.1 Designed by Junaid Brooks 12](#_Toc136184572)

[2.3.2 Designed by Ganeefa Sirkhotte 25](#_Toc136184573)

[3 DATA REQUIREMENTS 54](#_Toc136184574)

[Implementation Ready Class Diagram 54](#_Toc136184575)

# FUNCTIONAL REQUIREMENTS

## Analysis Use Case Model

## 1.2 Use Case Glossary and Responsibilities

|  |  |
| --- | --- |
| **Team Member Responsible: Junaid Brooks** | |
| **Use Case Id** | **Use Case Name** |
| A0600 | Accept Friend |
| A0700 | Deny Friend |
| D0100 | View Application |
| D0200 | Approve Influencer |
| D0300 | Reject Influencer |
| D0400 | View Petal Addition |
| D0500 | Approve Petal |
| D0600 | Reject Petal |
| **Queries/Reports** | |
| A0200 | Add Friend |
| A0900 | Follow Influencer |
| A1000 | View Friend Request |
| A1500 | View Friend |
| A1700 | View Account |
| A1800 | View Following |

|  |  |
| --- | --- |
| **Team Member Responsible: Ganeefa Sirkhotte** | |
| **Use Case Id** | **Use Case Name** |
| A0100 | Register Account |
| A0300 | Water Post |
| A0400 | Submit Influencer Application |
| A0500 | Answer Blooming Gratitude |
| A1300 | View Rosebud |
| B0100 | Add Petal |
| C0100 | Prompt Blooming Gratitude |
| C0200 | Post Rosebud |
| **Queries/Reports** | |
| A0800 | View Application Outcome |
| A1100 | View Gratitude History |
| A1200 | View Garden |
| A1400 | View Blooming Gratitude |
| A1600 | Select Flower |
| A1900 | View Water |
| B0200 | View Addition Outcome |

# UI Prototypes

## Team UI Guidelines

**Navigation & Control**

Navigation would be extremely simple. The main pages, the ‘My Friends’ and ‘Following’, would have a tab on the main screen which highlights which of the two the user is currently viewing. The app will remain consistent with placements, all pages would have a header which will consist of the logo, which is mainly centered and an account/ menu button, which gives the user access to all the other sections of the application. Navigation through the system and performing any task, such as posting or watering, would be intuitive. The application can further be navigated by means of the built in android system navigation.

Our aim is to keep it consistent with other modern day social media applications, and have it easy to learn. This navigation scheme ensures usability and an enjoyable user experience.



Figure 1 Main page tab

A picture containing handwriting, calligraphy, font, typography

Description automatically generated

Figure 2 Header

**Support For User Efficiency**

Each interface/ page would be designed with one main functionality and some may have the possibility of having other minor functions included in them.

**Recognition rather than recall**

The user-interface of our app is designed in such a way that it resembles and is consistent with most social media apps, making it something that is recognizable and familiar to users. Our additional features to regular social media apps would have its features, buttons and icons recognizable in a similar way to promote recognition. This consistency ensures that new social media users only need to learn the system once and users that have previously made use of other social media apps, would just have to learn the new features in our app.

**Aesthetic Design**

The app will have a fixed head, seen in figure 2, consisting of the menu/account button and the logo throughout all the pages on the app. Throughout the app we keep our font and font colour consistent and the same. The only font that differs is that of the logo. The application has a minimalistic design, with each page only having one main function. The app will have a warm pastel colour scheme, this can be seen in the apps background colour which is figure 3. As this app is about positivity we try not to use any colours associated with negativity. Buttons remain the same colour, a pastel blue which is displayed in figure 4, and style throughout the app. Tabs also remain the same colour and style throughout the app.



Figure 3 Background Colour



Figure 4 Buttons

**Error prevention and recovery**

The errors will be dealt with in code. A user would not be able to complete certain actions (creating an account, logging in and influencer application) if they have not filled in the details correctly. An error message will be displayed, similar to the one in figure 5, if the input is invalid. These error messages would be detailed enough for the user to know where their error lies.

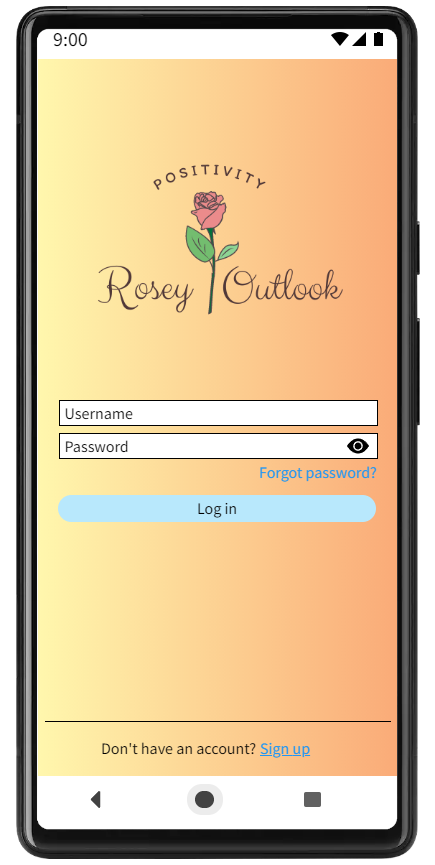
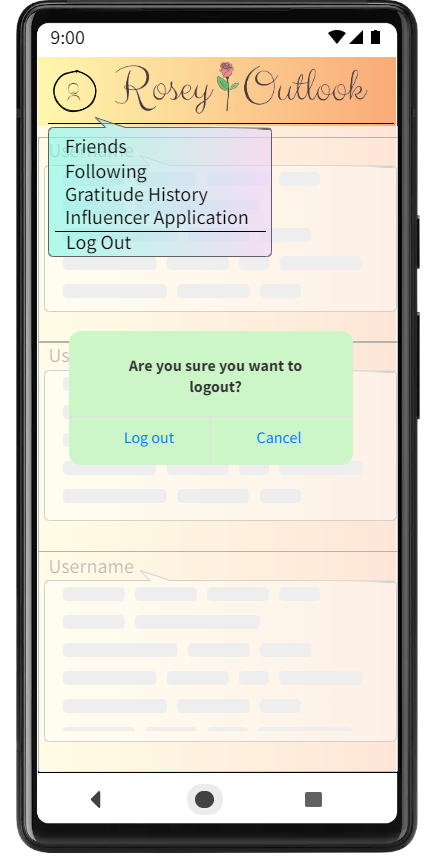
A picture containing text, font, typography, design

Description automatically generated

Figure 5 Error message

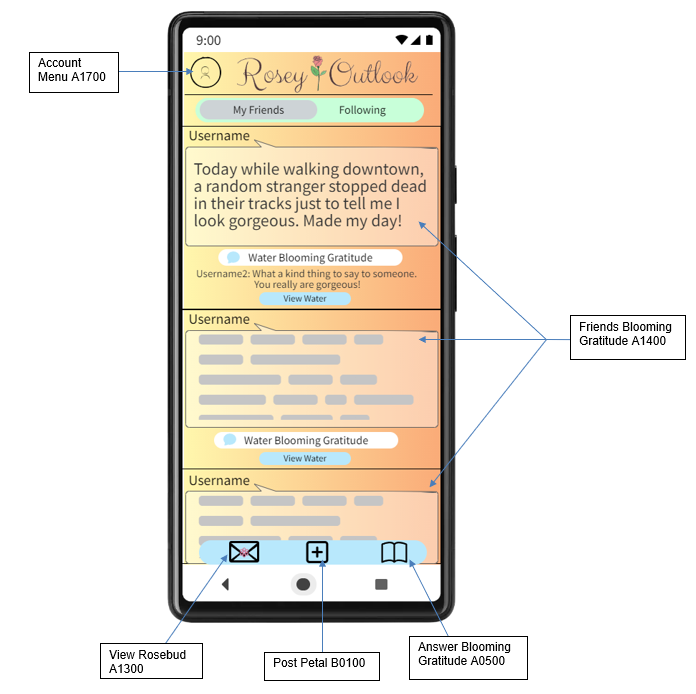
## Team Design for

### Login / Logout

### Dashboard / Landing Page / Main Page

#### Influencer’s Homepage



#### Public user’s homepage

A screenshot of a phone

Description automatically generated with low confidence

##### Admin

A screen shot of a phone

Description automatically generated with medium confidence

## UI Designs & Updated Analysis Use Case Narratives

### Designed by Junaid Brooks

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A0200 | Add Friend | |
| **Primary Business Actors** | | **Other participating Actors** |
| Public | | Influencer |
| **Description** | This use case describes the process of adding a new friend, or more specifically sending a friend request. | |
| **Pre-Conditions** | Public must have registered an account. Must be on the Friends tab of the Friends page. | |
| **Triggers** | Called from A1500 | |
| **Post-Conditions** | Friend request was sent. | |
| **Basic Flow of Events** | 1. User would click on the search/add friend button 2. User types in their friend’s username and hits the search button. 3. System checks if username exists. 4. User then selects friend’s account. 5. User then click on Add friend button. | |
| **Alternate Flow of Events** | 3.1 Check fails, system displays error.  3.2 Returns to step 2. | |
| **Initial UI Design** |  | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A0600 | Accept Friend | |
| **Primary Business Actors** | | **Other participating Actors** |
| Public | | Influencer |
| **Description** | Users will receive friend requests from other users, when they try to add them as a friend. User could then accept the friend request. | |
| **Pre-Conditions** | User has a registered account and has a pending friend request.  Public has to be in the Friends tab of the Friends page, which is invoked by A1500. | |
| **Triggers** | Invoked by A1000. | |
| **Post-Conditions** | A new friend was added. | |
| **Basic Flow of Events** | 1. User clicks the Add/ accept button. | |
| **Initial UI Design** |  | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A0700 | Reject Friend | |
| **Primary Business Actors** | | **Other participating Actors** |
| Public | |  |
| **Description** | A User receives a friend request from someone they do not wish to add, so they reject the request. | |
| **Pre-Conditions** | User has a registered account and have a pending friend request.  Public has to be in the Friends tab of the Friends page, which is invoked by A1500. | |
| **Triggers** | Invoked by A1000. | |
| **Post-Conditions** | User has denied the friend request. | |
| **Basic Flow of Events** | 1. User clicks the Deny/ reject button. | |
| **Initial UI Design** |  | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A0900 | Follow Influencer | |
| **Primary Business Actors** | | **Other participating Actors** |
| Public | | Influencer |
| **Description** | Follow a specific influencer to then have their flowers appear on your page. | |
| **Pre-Conditions** | User needs to have a registered account.  Public has to be in the Following tab of the Friends page. | |
| **Triggers** | Called from A1700. | |
| **Post-Conditions** | The User now follows the influencer. | |
| **Basic Flow of Events** | 1. User types in their influencers username and hits the search button. 2. System checks if username exists 3. User then selects influencer’s account. 4. User then click on follow button. | |
| **Alternate Flow of Events** | 2.1 Check fails, system displays error.  2.2 Returns to step 1. | |
| **Initial UI Design** |  | |

|  |  |
| --- | --- |
| **Use Case ID** | **Use Case Name** |
| A1000 | View Friend Request |
| **Actors** | |
| Public | |
| **Description** | This use case describes the process of viewing the friend requests that have been sent to a user. Both incoming and outgoing are visible to the user. |
| **Pre-Conditions** | User either needs to have a pending friend request or have sent out request that hasn’t been accepted nor rejected. |
| **Triggers** | Invoked by A1500 |
| **Post-Conditions** | The system displays the friend request that were sent to that user |
| **Basic Flow of Events** | 1. System displays a list of people that have sent friend requests to the user. There could also not be any new friend requests. 2. Select request to accept 3. Call A0600 |
| **Alternate Flow of events** | 2.1 Select request to reject  2.2 Call A0700 |
| **Initial UI design** |  |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A1500 | View Friend | |
| **Primary Business Actors** | | **Other participating Actors** |
| Public | | Influencer |
| **Description** | This use case describes the process of viewing the Friends section. | |
| **Pre-Conditions** | User must have an account. | |
| **Triggers** | Called from A1700. | |
| **Post-Conditions** | User’s friends are displayed. | |
| **Basic Flow of Events** | 1. The system displays the friends of the user. 2. User Selects to Add Friend 3. System calls A0200 | |
| **Alternate Flow of Events** | 2.1.1 Select Friend Requests  2.1.2 System calls A1000  2.2.1 User selects View Gratitude  2.2.2 System calls A1400 | |
| **Initial UI Design** |  | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A1700 | View Account | |
| **Primary Business Actors** | | **Other participating Actors** |
| Public | | Influencers |
| **Description** | This use case describes the process of viewing the Account menu | |
| **Pre-Conditions** | User must have a registered account. | |
| **Triggers** | User clicks Account button. | |
| **Post-Conditions** | A menu of options is displayed. | |
| **Basic Flow of Events** | 1. System displays a menu. 2. User selects Friends. 3. System calls A1500 | |
| **Alternate Flow of Events** | 2.1.1 User selects Following.  2.1.2 System calls A1800.  2.2.1 User selects Gratitude History.  2.2.2 System calls A1100.  2.3.1 User Selects Influencer Application,  2.3.2 System calls A0400.  2.4.1 User Selects My Garden.  2.4.2 System calls A1200 | |
| **Initial UI Design** |  | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A1800 | View Following | |
| **Primary Business Actors** | | **Other participating Actors** |
| Public | | Influencer |
| **Description** | This use case describes the process of viewing the Following section. | |
| **Pre-Conditions** | User must have an account. | |
| **Triggers** | Called from A1700. | |
| **Post-Conditions** | User’s following is displayed. | |
| **Basic Flow of Events** | 1.The system displays the influencers that the user is following.  2.User Selects to Follow Influencer  3.System calls A0900 | |
| **Alternate Flow of Events** | 2.1 User selects additional info.  2.2.1. User clicks View Gratitude  2.2.2 System calls A1400  2.3.1 User clicks on Influencer’s username  2.3.2 System calls A1200 | |
| **Initial UI Design** |  | |

|  |  |
| --- | --- |
| **Use Case ID** | **Use Case Name** |
| D0100 | View Application |
| **Actors** | |
| Admin | |
| **Description** | This use case describes the process of viewing all the current influencer applications that have been submitted (A0400). |
| **Pre-Conditions** | User has to be on an admin account. |
| **Triggers** | Admin clicks View Application button (Influencer Button). |
| **Post-Conditions** | A list of unprocessed influencer applications is displayed. |
| **Basic Flow of Events** | 1. The system displays all the influencer applications. 2. Admin selects to accept application 3. D0200 is called |
| **Alternate Flow of events** | 2.1 Admin selects to reject application  2.2 D0300 is called |
| **Initial UI design** |  |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| D0200 | Approve Influencer | |
| **Primary Business Actors** | | **Other participating Actors** |
| Admin | |  |
| **Description** | Admin makes a regular user a potential new influencer. | |
| **Pre-Conditions** | A user submitted an application to be an influencer. | |
| **Triggers** | Called from D0100 | |
| **Post-Conditions** | There is now a new influencer. | |
| **Basic Flow of Events** | 1. Admin will review the application. 2. Admin approves the influencer. 3. Return to step 1 in D0100 | |
| **Initial UI Design** |  | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| D0300 | Reject Influencer | |
| **Primary Business Actors** | | **Other participating Actors** |
| Admin | |  |
| **Description** | Admin denies a regular user becoming a new influencer. | |
| **Pre-Conditions** | A user submitted an application to be an influencer. | |
| **Triggers** | Called from D0100 | |
| **Post-Conditions** | Influencer application is rejected. | |
| **Basic Flow of Events** | 1. Admin would review the application. 2. Admin rejects the application. 3. Return to step 1 in D0100 | |
| **Initial UI Design** |  | |

|  |  |
| --- | --- |
| **Use Case ID** | **Use Case Name** |
| D0400 | View Petal Addition |
| **Actors** | |
| Admin | |
| **Description** | This use case describes the process of viewing all the pending petal additions that have been submitted by influencers (B0100). |
| **Pre-Conditions** | User has to be on an admin account |
| **Triggers** | Admin clicks the View Petal Addition button (Petals button) |
| **Post-Conditions** | A list of unprocessed petals is displayed |
| **Basic Flow of Events** | 1. The system displays all the Petal Additions that admin needs to approve/deny. 2. Admin selects to accept the petal addition 3. D0500 is called |
| **Alternate Flow of events** | 2.1 **Admin selects to reject the petal** addition  2.2 **D0600 is called** |
| **Initial UI design** |  |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| D0500 | Approve Petal | |
| **Primary Business Actors** | | **Other participating Actors** |
| Admin | |  |
| **Description** | The admin accept/ adds the new petal’s addition to the influencers garden. | |
| **Pre-Conditions** | Influencer has uploaded a petal that’s waiting for approval or to be denied. | |
| **Triggers** | Called from D0400 | |
| **Post-Conditions** | Influencer has a new petal in their garden. | |
| **Basic Flow of Events** | 1. Admin reviews the Petal. 2. Admin accepts the Petal. 3. System removes Petal from list to be reviewed by admin. 4. Return to step 1 of D0400 | |
| **Initial UI Design** |  | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| D0600 | Reject Petal | |
| **Primary Business Actors** | | **Other participating Actors** |
| Admin | |  |
| **Description** | The admin denies the new petal’s addition to the influencers garden. | |
| **Pre-Conditions** | Influencer has uploaded a petal that’s waiting for approval or to be denied. | |
| **Triggers** | Called from D0400 | |
| **Post-Conditions** | New flower is not added to the Garden. | |
| **Basic Flow of Events** | 1. Admin reviews the Petal. 2. Admin rejects the addition. 3. System removes Petal from list to be reviewed by admin. 4. Return to step 1 of D0400 | |
| **Initial UI Design** |  | |

### Designed by Ganeefa Sirkhotte

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A0100 | Register Account | |
| **Primary Business Actors** | | **Other participating Actors** |
| Public | |  |
| **Description** | This use case describes the process of registering a new account on the system. | |
| **Pre-Conditions** | Public must not already have an existing account with the same details. | |
| **Triggers** | Opening the application for the first time | |
| **Post-Conditions** | Public will have registered an account. | |
| **Basic Flow of Events** | 1. System displays the Register Account page. 2. User enters their preferred name. 3. User enters their desired username. 4. System checks that there is no existing account with the same username. 5. User enters their email address. 6. System checks that both email addresses are exactly the same. 7. User enters their password twice. 8. System checks that both passwords are exactly the same. 9. User clicks create account. 10. System checks that all details are filled in. 11. System displays homepage. | |
| **Alternate Flow of Events** | 4.1 Check fails, system displays error.  4.2 Returns to step 3 in basic flow.  6.1 Check fails, system displays error.  6.2 Returns to step 5 in basic flow.  8.1 Check fails, system displays error.  8.2 Return to step 7 in basic flow.  9.1 User clicks cancel.  9.2 All details filled in by the user is removed.  9.3 Return to step 1.  10.1 Check fails, system displays error.  10.2 Return to applicable step in the basic flow according to what was left out. | |
| **Initial UI design** | Click to create account  Click to view password  User fills in details  Required details  Clears details  4.1 (Username already exists)    Error message  6.1 (emails do not match)    Error message  8.1 (Password do not match)    Error message  10.1 (Not all details filled in)    Error message | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A0300 | Water Post | |
| **Primary Business Actors** | | **Other participating Actors** |
| Public | | Influencer |
| **Description** | This use case describes the process of a user commenting on a post, such as a Petal, Rosebud or Blooming Gratitude, after viewing it. | |
| **Pre-Conditions** | The user has to be viewing a post. | |
| **Triggers** | Called from A1300/A1400/A1600 | |
| **Post-Conditions** | The user will have left a comment on the post. This comment is then stored. | |
| **Basic Flow of Events** | 1. User types in their comment. 2. User presses the post button. 3. System displays their comment under the post. 4. Return to calling use case | |
| **Initial UI design** | User types in their comment  Post button    After the comment has been posted:    Comment displayed under post | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A0400 | Submit Influencer Application | |
| **Primary Business Actors** | | **Other participating Actors** |
| Public | |  |
| **Description** | A Public user would like to become an influencer and send an application to be an influencer. | |
| **Pre-Conditions** | Public needs to already have a registered account. | |
| **Triggers** | Called from A1700. | |
| **Post-Conditions** | An application to be an influencer has been submitted. | |
| **Basic Flow of Events** | 1. Display questionnaire 2. Fill in questionnaire 3. User clicks the submit button to Submit their application. 4. System checks if all required questions are answered. 5. System enable “View Outcomes” option | |
| **Alternate Flow of Events** | 3.1 User clicks cancel.  3.2 System returns user to most recently visited page.  4.1 Check fails, system displays an error message.  4.2 Return to Step 2. | |
| **Initial UI design** | Required questions  Returns user to homepage  User clicks submit  User fills in questionnaire  4.1 (Not all required questions answered)    Error message | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A0500 | Answer Blooming Gratitude | |
| **Primary Business Actors** | | **Other participating Actors** |
| Public | | Influencer |
| **Description** | This use case describes the process of a user answering the prompt for what they are grateful for or happy about on that day. | |
| **Pre-Conditions** | C0100 must have been invoked for the day (after 14:00)  The user must not have already answered Blooming Gratitude for that day. | |
| **Triggers** | Clicking the Blooming Gratitude button. | |
| **Post-Conditions** | The user will have posted what they are grateful for. | |
| **Basic Flow of Events** | 1. System displays the prompt. 2. User types in what they are grateful or happy for. 3. User presses the post button. 4. System checks if all required questions are answered. 5. The system displays Blooming Gratitudes that were posted by the user’s friends on that day and hides what their friends posted the previous day. | |
| **Alternate Flow of Events** | 3.1 User clicks cancel button.  3.2 System returns user to homepage.  4.1 Check fails, system displays an error message.  4.2 Return to Step 2. | |
| **Initial UI design** | Required question  User answers question  Returns user to homepage  User clicks Post  4.1 (Not all required questions answered)    Error message | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A1300 | View Rosebud | |
| **Primary Business Actors** | | **Other participating Actors** |
| Public | | Influencer |
| **Description** | This use case describes the process of viewing the Rosebud that was sent by the system in C0200. | |
| **Pre-Conditions** | User has an existing account. | |
| **Triggers** | User clicks Rosebud button/ User opens app for the first time since a new rosebud was posted (invoked by C0200). | |
| **Post-Conditions** | The user can view the new Rosebud for the day. | |
| **Basic Flow of Events** | 1. System displays the Rosebud as a pop up. 2. User selects Water Post 3. Call A0300 | |
| **Alternate Flow of Events** | 2.1.1 User clicks View Water  2.1.2 Call A1900  2.2.1 User clicks close.  2.2.2 System closes popup. | |
| **Initial UI design** | Closes popup  Calls A0300  Calls A1900 | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| B0100 | Add Petal | |
| **Primary Business Actors** | | **Other participating Actors** |
| Influencer | |  |
| **Description** | The influencer wants to add a new flower to their garden. They add what they would like to post, and it gets sent to the admins to get it reviewed(D0400). | |
| **Pre-Conditions** | User needs to be an influencer.  Influencer needs to be on their homepage. | |
| **Triggers** | Influencer clicks the post button to add a flower. | |
| **Post-Conditions** | Influencer has potentially added flower. | |
| **Basic Flow of Events** | 1. System displays questions. 2. User adds what they would like to post. 3. User clicks the post button. 4. System checks if all required questions are answered. 5. System enables Outcome option | |
| **Alternate Flow of Events** | 3.1 User clicks cancel button.  3.2 System returns user to homepage.  4.1 Check fails, system displays an error message.  4.2 Return to Step 2. | |
| **Initial UI design** | Required Question  User clicks post  User fills in details  Returns user to homepage  4.1 (Not all required questions answered)    Error message | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| C0100 | Prompt Blooming Gratitude | |
| **Primary Business Actors** | | **Other participating Actors** |
| Time | |  |
| **Description** | This use case describes the process of the system prompting the user to answer Blooming Gratitude. | |
| **Pre-Conditions** | At least one account has been registered. | |
| **Triggers** | This occurs automatically at 14:00. | |
| **Post-Conditions** | A new Blooming Gratitude prompt will be available to users to answer. | |
| **Basic Flow of Events** | 1. The system sends the Blooming Gratitude prompt to users. 2. The Blooming Gratitude Button on the homepage changes colour from black to pink. | |
| **Initial UI design** | Blooming Gratitude Button | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| C0200 | Post Rosebud | |
| **Primary Business Actors** | | **Other participating Actors** |
| Time | |  |
| **Description** | This use case describes the process of the system posting a Rosebud. | |
| **Pre-Conditions** | At least one account has been registered.  There is at least one Rosebud stored in the system. | |
| **Triggers** | This occurs automatically at 00:00. | |
| **Post-Conditions** | A new Rosebud will be available for users to view. | |
| **Basic Flow of Events** | 1. The system sends the Rosebud to users as a pop up. | |
| **Initial UI design** |  | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A0800 | View Application Outcome | |
| **Primary Business Actors** | | **Other participating Actors** |
| Public | |  |
| **Description** | User checks to see if their influencer application is approved or denied. | |
| **Pre-Conditions** | User would have had to applied to be an influencer. | |
| **Triggers** | User clicks on the become influencer button. | |
| **Post-Conditions** | User would have viewed the application outcome. | |
| **Basic Flow of Events** | 1. User would then click on the View Outcomes option. 2. The application outcome will be displayed. 3. The application was successful, and the user clicks the Become Influencer button. 4. The user is now switched over to the influencer version of the app. | |
| **Alternate Flow of Events** | 3.1.1 The application was unsuccessful.  3.1.2 The user clicks the Mark as read button.  3.1.3 System disables “View Outcome” option.  3.2.1 The application is still being processed. | |
| **Initial UI design** | 1. Successful application     Normal user account becomes Influencer account  Outcome displayed  3.1.1 Unsuccessful application    User acknowledges outcome and does not need to view it again  3.2.1 Unprocessed application | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A1100 | View Gratitude History | |
| **Primary Business Actors** | | **Other participating Actors** |
| Public | | Influencer |
| **Description** | This use case describes the process of a user being able to view all the Blooming Gratitude posts that they have made. | |
| **Pre-Conditions** | User has answered at least one Blooming Gratitude (A0500). | |
| **Triggers** | Called from A1700. | |
| **Post-Conditions** | Every answer that the user has posted to Blooming Gratitude is displayed. | |
| **Basic Flow of Events** | 1. The system displays the user’s entire history of answers to Blooming Gratitude(by default will be sorted by date). 2. User clicks filter by. 3. System displays filtering options. 4. User selects options. 5. System displays Blooming Gratitude posts within the selected date range. | |
| **Initial UI design** | Click to display filtering options    Filtering options    User selects option  Filter has been applied | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A1200 | View Garden | |
| **Primary Business Actors** | | **Other participating Actors** |
| Public | | Influencer |
| **Description** | This use case describes the process of viewing the garden of an influencer. | |
| **Pre-Conditions** | The user is viewing posts under their influencers section of the application, or they are on the Following page. | |
| **Triggers** | User clicks the Influencer’s username, invoked by A1800/A1400. | |
| **Post-Conditions** | The garden of a specific influencer will be displayed. | |
| **Basic Flow of Events** | 1. The system displays the Garden of the selected influencer. 2. User clicks Select Flower 3. A1600 is invoked. | |
| **Alternate Flow of Events** | 2.1.1 User clicks Water Petal.  2.1.2 System calls A0300  2.2.1 User clicks View Water.  2.2.2 System calls A1900. | |
| **Initial UI design** | These petals in this block belong to the same Flower (Advice)  Indicates which Flower these petals belong to  Calls A1900  Calls A0300  Calls A1600 | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A1400 | View Blooming Gratitude | |
| **Primary Business Actors** | | **Other participating Actors** |
| Public | | Influencer |
| **Description** | This use case describes the process of viewing the Blooming Gratitude posted by all the user’s friends or influencers they are following. | |
| **Pre-Conditions** | The user must have already answered Blooming Gratitude for that day (A0500). | |
| **Triggers** | User clicks the Friends or Following tab on their homepage. / User clicks View Gratitude from the Friends (invoked by A1500) or Following (invoked by A1800) pages | |
| **Post-Conditions** | The user can view the Blooming Gratitude answers. | |
| **Basic Flow of Events** | 1. The system displays the Blooming Gratitude answers of either the user’s friends or those they follow. 2. User selects one to water 3. Call A0300 | |
| **Alternate Flow of Events** | * 1. System displays the Blooming Gratitude of the selected user as a pop up.   2. Return to step 2.   2.1.1 User selects View Water  2.1.2 Call A1900  2.2.1 User clicks Close button.  2.2.2 System closes pop up and returns to calling use case.  2.3.1 User clicks on Influencer’s username.  2.3.2 System calls A1200. | |
| **Initial UI design** | Triggered by clicking Friends tab of the home page    Calls A1900  Calls A0300  User’s friends’ Blooming Gratitude  Triggered by clicking Following tab of the home page    Calls A1200  Calls A0300  Influencer’s Petal  Influencer’s Blooming Gratitude  Triggered by A1500/A1800    Closes popup  Calls A1900  Calls A0300 | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A1600 | Select Flower | |
| **Primary Business Actors** | | **Other participating Actors** |
| Public | | Influencer |
| **Description** | The user would go onto an influencers page and select the flower the user would like to view. | |
| **Pre-Conditions** | User must have a registered account. | |
| **Triggers** | User is viewing an influencer’s Garden invoked by A1200. | |
| **Post-Conditions** | Viewing the petals in the Influencers flower. | |
| **Basic Flow of Events** | 1. Systems displays which flowers are available in the specific influencer’s Garden. 2. User would select a flower to view. 3. User would click on the flower. (User clicks apply filter) 4. User would now be able to view all the petals within the flower. 5. Select to Water Post 6. Call A0300 | |
| **Alternate Flow of Events** | 5.1 User selects View Water  5.2 Call A1900 | |
| **Initial UI design** | Calls A1900  Calls A0300  Flowers that are in this Influencer’s Garden    User selects a Flower  After filter has been applied | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A1900 | View Water | |
| **Primary Business Actors** | | **Other participating Actors** |
| Public | | Influencer |
| **Description** | This use case describes the process of a user viewing comments on a post, such as a Petal, Rosebud or Blooming Gratitude, after viewing it. | |
| **Pre-Conditions** | The user has to be viewing a post. | |
| **Triggers** | Called from A1300/A1400/A1600 | |
| **Post-Conditions** | The user will see all the comments that have been left on the post. | |
| **Basic Flow of Events** | 1. System displays comments under the post | |
| **Initial UI design** | Hides comments again  Comments | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| B0200 | View Addition Outcome | |
| **Primary Business Actors** | | **Other participating Actors** |
| Influencer | |  |
| **Description** | Influencers would like to see if their new petal is approved(D0500) or not(D0600). | |
| **Pre-Conditions** | Influencer needed to submit a petal for review.  Influencer needs to be on their homepage. | |
| **Triggers** | Clicking the post button. (Called from B0100) | |
| **Post-Conditions** | Knowing the status of the petal. | |
| **Basic Flow of Events** | 1. Select the option to view petal addition outcome. 2. System displays list of the outcomes of all the petal additions. 3. User clicks Mark as read button for a specific outcome. 4. System removes this petal addition’s outcome from the list of outcomes. | |
| **Initial UI design** | These petals in this block belong to the same Flower (Advice)  User acknowledges outcome and does not need to view it again  Indicates which Flower these petals belong to  Addition outcome has been removed after the user clicked Mark as read | |

# DATA REQUIREMENTS

## Implementation Ready Class Diagram

|  |  |
| --- | --- |
| **Entity** | **Entity Attributes** |
| Public | **Username <pk>**, prefferedName, email, password, Type, profilePic |
| Rosebud | **R\_ID <pk>**, Affirmation, TimesUsed, DateScheduled |
| Blooming Gratitude | **BG\_ID <pk>,** Date, Grateful, *Username <fk>* |
| Petal | **P\_ID <pk>**, *Username <fk>*, Content, *F\_ID <fk>*, Date |
| PetalSubmission | **PS\_ID <pk>**, *Username <fk>*, Content, *F\_ID <pk>* |
| Flower | **F\_ID <pk>**, Category |
| Garden | **Username <pk>**, *F\_ID <fk>* |
| Influencer | **Username <pk>** |
| InfluencerApplication | **I\_ID <pk>**, *Username <fk>,* PotentialPetal, Reason |
| InfluencerApplicationFlower | ***I\_ID <pk><fk>, F\_ID <pk><fk>*** |
| SuggestedFlower | **SF\_ID <pk>**, *I\_ID <fk>*, Suggestion |
| Friendship | **Friendship\_ID <pk>,** *F1\_Username <fk>,* *F2\_Username* *<fk>*, StartDate |
| FriendInvite | **FI\_ID <pk>**, *fromUsername <fk>, toUsername <fk>* |
| Following | **Following\_ID,** Username, InfluencerUsername |
| Admin | **A\_ID <pk>,** Password, FirstName, Surname |
| WaterBloomingGratitude | **WBG\_ID <pk>,***BG\_ID <fk>,**Commentor <fk>, Poster <fk>,* Comment |
| WaterRosebud | **WR\_ID <pk>,***R\_ID <fk>,**Commentor <fk>, Poster <fk>,* Comment |
| WaterPetal | **WP\_ID <pk>*,*** *P\_ID <fk>,**Commentor <fk>, Poster <fk>,* Comment |

**A picture containing text, font, white, graphics

Description automatically generatedDepartment of Computing Sciences**

**Plagiarism Declaration - Elaboration Document**

**Module code:** WRRV301

**We,**

|  |  |  |
| --- | --- | --- |
| **Name** | **Student Number** | **Contribution % [Total = 100]** |
| **Ganeefa Sirkhotte** | **224078321** | 50 |
| **Junaid Brooks** | **223126756** | 50 |
|  |  |  |
|  |  |  |

**hereby declare that this submission is our own, original work.**

We further declare that:

1. No part of this submission has been copied from another person/group,

2. We **did / did not** work with another person/group on this submission,

3. We acknowledged all consulted sources in the text and submitted a list of references, and

4. Parts without references are entirely our own work

5. That we have all equally contribute to the work or as indiciated in the contribution % above.

*(Delete any of the above if not applicable)*

We understand that, should this declaration be false, we may be charged with academic misconduct and/or plagiarism and that a disciplinary hearing may be brought against us.

1. A picture containing symbol

   Description automatically generatedSIGNED: ............................................... DATE......28/05/2023.......................
2. SIGNED: ............................................... DATE .......28/05/2023.....................