|  |  |
| --- | --- |
| Author | Thanh Vo |
| Date | 26-06-2018 |
| Version | 3.35 |
| Status | Reviewing |
| Project | Give and Receive Application |

**SRS\_GiVE\_AND\_RECEIVE APP**

**TABLE OF CONTENTS**

[1 INTRODUCTION 5](#_Toc519079207)

[1.1 Purpose 5](#_Toc519079208)

[1.2 Scope 5](#_Toc519079209)

[2 REFERENCED DOCUMENTS 5](#_Toc519079210)

[2.1 Controlling Documents 5](#_Toc519079211)

[2.2 Controlled Documents 5](#_Toc519079212)

[2.3 Background Information 5](#_Toc519079213)

[3 ACTORS 7](#_Toc519079214)

[4 OVERVIEW 7](#_Toc519079215)

[5 FEATURES 8](#_Toc519079216)

[5.1 Device Orientation 8](#_Toc519079217)

[5.2 Startup 8](#_Toc519079218)

[5.2.1 User Interface 8](#_Toc519079219)

[5.2.2 Business Logic 8](#_Toc519079220)

[5.3 Main 8](#_Toc519079221)

[5.3.1 User Interface 8](#_Toc519079222)

[5.3.2 Business Logic 8](#_Toc519079223)

[5.3.3 User Cases 9](#_Toc519079224)

[5.4 Login 9](#_Toc519079225)

[5.4.1 User Interface 9](#_Toc519079226)

[5.4.2 Business Logic 9](#_Toc519079227)

[5.4.2.1 Login using Facebook/Google account 9](#_Toc519079228)

[5.4.2.2 Remember credentials 10](#_Toc519079229)

[5.4.2.3 Lost internet connection 10](#_Toc519079230)

[5.4.3 Use Cases 10](#_Toc519079231)

[5.5 Create/Modify/Delete item in giver’s list 11](#_Toc519079232)

[5.5.1 User Interface 11](#_Toc519079233)

[5.5.2 Business Logic 11](#_Toc519079234)

[5.5.2.1 Create new item 11](#_Toc519079235)

[5.5.2.2 Post items 11](#_Toc519079236)

[5.5.2.3 Modify items 11](#_Toc519079237)

[5.5.2.4 Delete items 12](#_Toc519079238)

[5.5.3 Use Cases 12](#_Toc519079239)

[5.6 Claim items to “Người Nhận” 13](#_Toc519079240)

[5.6.1 User Interface 13](#_Toc519079241)

[5.6.2 Business Logic 13](#_Toc519079242)

[5.6.3 Use Cases 13](#_Toc519079243)

[5.7 Request available items 14](#_Toc519079244)

[5.7.1 User Interface 14](#_Toc519079245)

[5.7.2 Business Logic 14](#_Toc519079246)

[5.7.3 Use Cases 14](#_Toc519079247)

[5.8 Category 15](#_Toc519079248)

[5.8.1 User Interface 15](#_Toc519079249)

[5.8.2 Business Logic 15](#_Toc519079250)

[5.8.3 Use cases 15](#_Toc519079251)

[5.9 Remove violated items, ban users 15](#_Toc519079252)

[5.9.1 User Interface 15](#_Toc519079253)

[5.9.2 Business Logic 15](#_Toc519079254)

[5.9.2.1 Manage a list of violated Items 15](#_Toc519079255)

[5.9.3 Use cases 16](#_Toc519079256)

[5.10 Create/Modify/Delete category 17](#_Toc519079257)

[5.10.1 User Interface 17](#_Toc519079258)

[5.10.2 Business Logic 17](#_Toc519079259)

[5.11 Give comments/feedbacks 18](#_Toc519079260)

[5.11.1 User Interface 18](#_Toc519079261)

[5.11.2 Business Logic 18](#_Toc519079262)

[5.11.3 Use cases 18](#_Toc519079263)

[5.12 Create ranking system for “Người Cho” 18](#_Toc519079264)

[5.12.1 User Interface 18](#_Toc519079265)

[5.12.2 Business Logic 19](#_Toc519079266)

[5.13 Rank the item/post 19](#_Toc519079267)

[5.13.1 User Interface 19](#_Toc519079268)

[5.13.2 Business Logic 19](#_Toc519079269)

[5.14 Report the item/post 19](#_Toc519079270)

[5.14.1 User Interface 19](#_Toc519079271)

[5.14.2 Business Logic 19](#_Toc519079272)

[5.15 Create/Update/Delete Post 19](#_Toc519079273)

[5.15.1 User Interface 20](#_Toc519079274)

[5.15.2 Business Logic 20](#_Toc519079275)

[5.15.2.1 Create a post 20](#_Toc519079276)

[5.15.2.2 Edit a post 20](#_Toc519079277)

[5.15.2.3 Delete a post 20](#_Toc519079278)

[6 Non-functional Requirements 22](#_Toc519079279)

[6.1 Operating System Support 22](#_Toc519079280)

[7 ANALYZE AND DESIGN 22](#_Toc519079281)

[7.1 Activity Diagram 22](#_Toc519079282)

[7.2 Class Diagram 22](#_Toc519079283)

[7.3 Database 22](#_Toc519079284)

[8 APPENDIX A: [TBD] 22](#_Toc519079285)

**CHANGE HISTORY**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Description** |
| 0.1 | 03-10-2016 | Robert Hendriksen | Initial version |
|  |  |  |  |

# INTRODUCTION

## Purpose

The purpose of this document is to describe the requirements and specification for the Scientific Workflows application on iPad

## Scope

This document is intended for anyone in direct relation to the Give and Receive application.

# REFERENCED DOCUMENTS

## Controlling Documents

This document is dependent on the following documents:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Id | Document | Author | Version | Date |
|  | FEI Workflows – Mockups.pdf | Author Name |  | Aug 21 2016 |
|  | Database\_Design V2.1.jpeg | Author Name |  | Aug 21 2016 |
|  | XML file structure 1.4.docx | Author Name |  | Oct 06 2016 |
|  | UI propsosal\_0033645 UI design iPhone.pdf | Author Name |  | Apr 13 2018 |

## Controlled Documents

|  |  |  |
| --- | --- | --- |
| UI Design | Ky Pham | Sep 26 2016 |
| Reworked UI Design | Ky Pham | Feb 2 2017 |
|  |  |  |

## Background Information

The following documents are relevant to the context of the document but do not affect the contents in a direct way:

[SQS]: Sioux Quality System

Version 3.5  
August 7, 2012

# ACTORS

The following table gives an overview of the different human and non-human actors that interact with the system. The name between parentheses is the formal name further used in this document.

|  |  |
| --- | --- |
| **Actor** | **Description** |
| Operator | Who execute a workflow step by step with different instrument. |
| S3 | A server which holds all available workflows and its relevant files, results, images, and videos. |

# OVERVIEW

Description for overview of application

# FEATURES

## Device Orientation

**Give and Receive App** supports both Portrait and Landscape mode for supported devices:

iPad and iPhone with the ***minimum*** ***iOS 11.0***, multi-touch display with LED backlight and IPS technology.

And android too… need to have Mr Thanh’s slide to write this information.

* For ***iPad***: the minimum model supported is **iPad Air 2**.
* For ***iPhone***: the minimum model supported is **iPhone 5s** ***(Performance issues are acceptable)*** and the ***Standard Testing Device*** is **iPhone 6s**. The application also supports ***iPhone X***

## Startup

### User Interface

For Iphone

[TBD]

For Android:

[TBD]

### Business Logic

Splash screen is the first screen to show whenever user opens the application.

Splash screen remains during loading progress and close after the loading finished.

Once the app is finished loading, it will show the login screen.

## Main

Main screen is shown after the application is loaded successfully.

Guests and users can see all the posts about items to be given.

### User Interface

*[TBD]*

### Business Logic

* Main screens is loaded after the application is loaded successfully. In main screen, users can see all the posts of “Người Cho”.
* Besides that, users can use filter to classify the posts to see what they concern.
* The value of filters are pre-defined by dev team.
* If user apply filters, the result will be shown based on the filter, if not, the result will be all of the posts available on our system.

### User Cases

|  |  |  |
| --- | --- | --- |
| Use Case | UC1: Give and Receive App allows users to see all the items on the main screen. | |
| Actors | **Guests** | **User** |
| Pre-conditions | N/A | |
| Post-conditions | N/A | |
| Main success scenario | 1. The user launches Give and Receive app 2. Give and Receive app displays Launching screen 3. Give and Receive app displays Main screen 4. The user see all the posts in a list, scroll down and up to see the posts | |
| Alternative scenario 1 | 1. Launch Give and Receive app from background    1. Give and Receive app displays the last screen | |
| Business Rules |  | |
| Other requirements | Time of launching app is as short as possible (within 3s loading for items filled the screen) | |
| Open issues |  | |

## 

## Login

### User Interface

For Iphone

*[TBD]*

For Android:

*[TBD]*

### Business Logic

#### Login using Facebook/Google account

There are two buttons on the login screen. One for login using Facebook account and one for login using Google account.

* If the user want to log in using their Facebook account, they just touch on “Login using Facebook account” button.
  + If the device is not yet log in to Facebook, a new screen will appear base on browser app on the user phone, require the user to log in to Facebook. Then a new screen appear to confirm with the user some privacy problem like “Allow Cho Và Nhận application to get your information like Avatar, Name, DOB, email,...”. If the user agree, start checking for updates and then go to the main screen.
  + If the device is logged in to Facebook, skip the “log in to Facebook” step, redirect to the confirm step.

If the user want to create an account base on their Google account. Do the same thing as “Login using Facebook account” but with Google account.

#### Remember credentials

It is possible to remember the credentials. In this case the credentials information (Facebook/Google account that linked with app) will be used for next login purpose. User will not need to “Sign in with Facebook” or “Sign in with Google” again, the App will be logged in automatically.

#### Lost internet connection

If the user’s phone lost internet connection, show the “Waiting to reconnect to the internet” box and then wait for reconnect to the internet.

After 1 minutes, if the user’s phone is not yet connect to the internet, show the “Failed to connect to the internet” box, and then closed the app.

If the app re-connected to the internet, close the error box.

### Use Cases

|  |  |  |
| --- | --- | --- |
| Use Case | UC1: Login using Facebook/Google account | |
| Actors | **Guest** | Operator **User want to use the app within their existing account** |
| Pre-conditions | The user touch the button “Login using Facebook account” or “Login using Google account” on Login Screen | |
| Post-conditions | The user logged in successfully | |
| Main success scenario | 1. The user successfully logged in to Facebook. 2. The user touch “I agree” button on the “Confirm privacy” screen 3. Star check for updates. 4. Redirect to the main screen. | |
| Alternative scenario 1 | 1. The user touch “I don’t agree” button on the “Confirm privacy” screen. 2. Redirect to the login screen. | |
| Alternative scenario 2 | 1. The user failed to login with Facebook. 2. Wait until the user successfully logged in to Facebook. | |
| Business Rules |  | |
| Other requirements | Time of launching app is as short as possible (3 - 5s) | |
| Open issues |  | |

## Create/Modify/Delete item in giver’s list

Giver Item list screen is shown after user click “My items” icon.

There are 3 tabs in the tab bar of this screen:

- Đang cho: items that are available on platform, all people can see it and request to receive it  
- Kho hàng: items that giver uploaded but not ready to give away  
- Đã cho: items that giver has given to receiver

### User Interface

[TBD]

### Business Logic

Giver Item list screen is shown after user click “My items” icon.

User swipes left or right to move to other tabs

#### Create new item

User can create a new item by filling all required information :

* take a picture of the item or upload the picture in their gallery
* name of the item that they want to give away
* the item's description

After that, there are some options for the user :

* post the item to the platform. The item's status is changed to “Đang cho”
* save the item to their private list and modify/edit/post it later. The item's status is changed to “Kho hàng”
* cancel

#### Modify items

User can’t modify items in “Đã cho” list . For the items in “Đang cho” list, after giver modify it, a notification will be sent to all the receiver that are requesting this item.

#### Delete items

User can only delete items that have no request. For the items in “Đang cho” list, the giver must reject all the requests in order to delete the item.

### Use Cases

|  |  |  |
| --- | --- | --- |
| Use Case | UC1: Give and Receive App allows Givers to Create a new item | |
| Actors | **Primar Primary** | Operat **User** |
| Pre-conditions | The user have logged in successfully | |
| Post-conditions | The user create a new item successfully | |
| Main success scenario | 1. The user click Create new item’s icon 2. The Give and Receive app displays Create Item screen 3. User fill in all the required information 4. The user click Post button to post the item to public. | |
| Alternative scenario 1 | Step 4. The user click Save button to save the item to the private list | |
| Alternative scenario 2 | Step 4. The user click Cancel button | |
| Alternative scenario 3 | Step 4. The Give and Receive app notifies required fields are missing. | |

|  |  |  |
| --- | --- | --- |
| Use Case | UC1: Give and Receive App allows Givers to Modify items | |
| Actors | **Primar Primary** | Operat **User** |
| Pre-conditions | The user have logged in successfully | |
| Post-conditions | The user post a new item successfully | |
| Main success scenario | 1. The giver choose items to post 2. The giver fill in required information 3. The giver submit to post | |
| Alternative scenario 1 | Step 3. The user click Cancel button to cancel the progress | |
| Alternative scenario 2 | Step 3. The Give and Receive app notifies required fields are missing. | |

|  |  |  |
| --- | --- | --- |
| Use Case | UC1: Give and Receive App allows Givers to Delete items | |
| Actors | **Primar Primary** | Operat **User** |
| Pre-conditions | The giver is in “Kho Hàng” or “Đang Cho” or “Đã Cho” tab | |
| Post-conditions | The user post a delete items successfully | |
| Main success scenario | 1. The giver choose items to delete 2. The giver submit to delete | |
| Alternative scenario 1 | Step 2. The giver cancel the progress | |
| Alternative scenario 2 | Step 3. The Give and Receive app notifies required fields are missing. | |

## Claim items to “Người Nhận”

The screen help “Người Cho” manage request from “Người Nhận” for items on “Người Cho”‘s post.

### User Interface

[TBD]

### Business Logic

* Each request just for an item. “Người Nhận” can also choose many items to request as the same time. After that, system will spread automatically into n different requests for n different items
* After “Người Nhận” request to receive a items, the requests will be sent to “Người Cho”
* “Người Cho” will receive n notification for n items which “Người Nhận” requested.
* “Người Cho” will check information of per request and then they will decide who will take those items and quantity.

### Use Cases

|  |  |
| --- | --- |
| Use Case | UC1: “Người cho” claim item to “Người Nhận”. |
| Actors | **User** |
| Pre-conditions | The user have logged in successfully |
| Post-conditions | N/A |
| Main success scenario | 1. “Người Cho” presses notification for each request. 2. “Cho và nhận” app displays request detail for this item. 3. “Người cho” adjust quantity and presses the Accept button. 4. “Cho và nhận” displays list of requests again to user 5. “Cho và nhận” app changes item’s state to be given or updates item’s quantity and the post. 6. “Cho và nhận” app sends notification to “Người nhận” requesting the item on the pos to let them know the changes about items and quantity. (Only “Người nhận” who requests for this item receive notification). 7. A message will be sent to chosen “Người Nhận” to notify the confirmation of “Người Cho” and telephone number of “Người Cho” to contact and receive the items. |
| Alternative scenario 1 | 1. “Người Cho’’ launch request management in app. |
| Alternative scenario 2 | 1. “Người Cho” presses Decline button.    1. The request is removed. |

## Request available items

The screen helps “Người Nhận” making a request to receive items from “Người Cho”

### User Interface

[TBD]

### Business Logic

* “Người Nhận” see the post and items they need on it. They press button “Create new request” inside the post.
* “Người Nhận” has to fill all mandatory fields such as: telephone number, item name, quality, reason... in request form to send to “Người Cho”.
* As for telephone number, “Người Nhận” can reuse their default telephone number on the profile, or they can fill the new number.
* As for item name, “Người Nhận” only choose the name of items in the post (Multichoice with predefined value)
* As for item quality, “Người Nhận” can choose the number (Maximum is the max value of this item in the post).
* If one of mandatory fields is empty, an error message will be shown to “Người Nhận”.
* “Người Nhận” can’t change the sent requests.

### Use Cases

|  |  |  |
| --- | --- | --- |
| Use Case | UC1: Create Request And Send to “Người Cho” | |
| Actors | **User** |  |
| Pre-conditions | The user have logged in successfully | |
| Post-conditions | The request created successfully | |
| Main success scenario | 1. “Người Nhận” loading ‘Create Request’ screen for a post 2. “Người Nhận” fill the form with mandatory fields. 3. “Người Nhận” press ‘Submit’ button 4. “Người Nhận” will be navigated into ‘Request Management’ screen | |
| Alternative scenario 1 | 1. “Người Nhận” does not fill all mandatory fields 2. “Người Nhận” press “Gửi” button    1. An error message will be shown to ‘Nguoi Nhan’ | |
| Alternative scenario 2 | 1. “Người Nhận” press “Hủy” button    1. “Người Nhận” will be navigated into the Post screen | |
| Other requirements |  | |
| Open issues |  | |

## Category

This screen help user to choose the category of items they want to see.

### User Interface

[TBD]

### Business Logic

* Category values is defined by admin.
* User press a picture of a category, the main screen will be shown with filter for this category.

### Use cases

|  |  |  |
| --- | --- | --- |
| Use Case | UC1: Choose the category | |
| Actors | **User** |  |
| Pre-conditions | N/A | |
| Post-conditions | The main screen will be shown with category filter. | |
| Main success scenario | 1. “Người Nhận” launch Category screen. 2. “Người Nhận” press a particular button (value) in screen 3. Main screen is shown with result filtrated by value of category. | |
| Other requirements |  | |
| Open issues |  | |

## Remove violated items, ban users

### User Interface

[TBD]

### Business Logic

#### Manage a list of violated Items

In CMS, violated post are displayed as a list. Each row includes these information:

* Index
* Post’s link
* Number of report
* Detail of reports (a link)
* Name of violated user is accompanied by number of warnings
* Block User button (block violated user)
* Warn User button
* Delete Post button
* Solved or unsolved state

Admin can click Post’s link to see detail of the post, then checking the violated item.

Admin can send warning notification to user or ban the user if number of warning more than 1.

Admin can delete violated post, all items that was attached to this post will be deleted as well.

### Use cases

|  |  |  |
| --- | --- | --- |
| Use Case | UC1: Send warning notification to user | |
| Actors | **Admin** | Operat |
| Pre-conditions | Login to CMS successfully | |
| Post-conditions | N/A | |
| Main success scenario | 1. Admin clicks “Warn User” button 2. System displays confirm box 3. Admin clicks OK button 4. System sends a warning notification to user’s phone and updates number of warnings of that user 5. “Warn User” button changes to “Warned” button (disable button) 6. Unsolved state changes to Solved | |
| Alternative scenario 1 | 1. User clicks Cancel button    1. Close confirm box | |

|  |  |  |
| --- | --- | --- |
| Use Case | UC2: Block user | |
| Actors | **Admin** |  |
| Pre-conditions | Login to CMS successfully | |
| Post-conditions | N/A | |
| Main success scenario | 1. Admin clicks “Block User” button 2. System displays confirm box 3. Admin clicks OK button 4. System updates user’s state to blocked 5. “Block User” button changes to “Blocked” button (disable button) 6. Unsolved state changes to Solved | |
| Alternative scenario 1 | Step 3. User clicks Cancel button  8.1.1 Close confirm box | |

|  |  |  |
| --- | --- | --- |
| Use Case | UC3: Delete post | |
| Actors | **Admin** |  |
| Pre-conditions | Login to CMS successfully | |
| Post-conditions | N/A | |
| Main success scenario | 1. Admin clicks “Delete” button 2. System displays confirm box 3. Admin clicks OK button 4. System removes the post and all items was attached to this post 5. System sends notification to user (information of post and all items deleted) 6. “Delete” button changes to “Deleted” button (disable button) 7. Unsolved state changes to Solved | |
| Alternative scenario 1 | Step 3. User clicks Cancel button  8.1.1 Close confirm box | |

## Create/Modify/Delete category

### User Interface

[TBD]

### Business Logic

In CMS, categories are displayed as a list. Each row includes these information:

* Index
* Category’s name
* Description
* Edit button
* Delete button

And Create button is above the list.

Admin can modify, delete per category. After deleting, category will be removed from the list, however, it still exists in database.

Admin can create a new category by pressing Create button, then fill in all necessary fields (Name, Description) and click OK button afterward.

## Give comments/feedbacks

### User Interface

[TBD]

### Business Logic

User can give comments, feedback by filling what they think about the app, what is the advantages and disadvantages of the app, what do they think to improve the app better,... to the shout box and submit it.

### Use cases

|  |  |  |
| --- | --- | --- |
| Use Case | UC1: Create feedback/ comment | |
| Actors | **User** | Operat |
| Pre-conditions | N/A | |
| Post-conditions | The feedback has sent successfully | |
| Main success scenario | 1. User click “Give feedback” icon 2. User fill in comments/ feedback 3. User submit the form | |
| Alternative scenario 1 | Step 3. User cancel | |
| Alternative scenario 2 | Step 3. The Give and Receive app notifies message field are missing. | |

## Create ranking system for “Người Cho”

### User Interface

[TBD]

### Business Logic

The rank of “Người cho” is defined base on the number of items that the user has been given, the rate of the the post, the number of posts that user has been reported,.. There is a list that show top 50 generous givers.

There is a system to calculate the number of things that a user “đã cho” and then depend on that information to rank that person.

Each rank will have a label come with a special color for their name.

## Rank the item/post

### User Interface

[TBD]

### Business Logic

Below each post, together with detail and action buttons for the users, there is a small area contain 5 star for the user to rank for the thing that the “Người Cho” want to “cho”

User can rank the items that are useful for other people can easier recognize it. User can rank from 1 star to 5 stars.

Rank of the item will display base on the number of bright stars.

Rank of the item will be calculate by the average score of those who have evaluated.

## Report the item/post

### User Interface

[TBD]

### Business Logic

User can report the items/posts that are violated by click the report button and fill the reason to the form. The report will be sent to the admin for handling.

The report button located within the action button on each post.

If the user want to report some post or some item. They will have to click on report button, then a new screen appear require the reporter to input the reason why they want to report that post/item. And submit it.

## Create/Update/Delete Post

This screen help user manage their posts.

### User Interface

[TBD]

### Business Logic

#### Create a post from “Kho hang” list

User can select items with the quantity from their Store and then presses “Create Post” button to create a post.

User can write description for the posts

#### Edit a post

User can edit their post, we have some situations:

* Edit post’ s description -> send a notification to all “Người nhận” requested
* Edit item that was not requested in post -> don’t send a notification
* Edit item that was requested in post -> send a notification to “Người nhận” requested that item

#### Delete a post

User can delete their post, we have some situations:

* Delete post that does not have any request -> don’t send a notification
* Delete post that was requested in post -> Handle all the request (Accept or Decline) and send a notification to “Người nhận” requested that item.
  + 1. **Use Cases**

|  |  |  |
| --- | --- | --- |
| Use Case | UC1: User Create post | |
| Actors | **User** | Operat |
| Pre-conditions | The user have logged in successfully | |
| Post-conditions | The post has been created successfully | |
| Main success scenario | 1. User pick items and define quantity (Default is max number of this item). 2. User click “+” icon in items list 3. User fill the mandatory fields: title, description, location, categories 4. User press “Create” button. 5. The post and items will be public to other users. | |
| Alternative scenario 1 | 1. User remove chosen items for the post    1. removed items disappear from the view. | |
| Alternative scenario 2 | 1. User does not fill all mandatory fields and press “Create”    1. an error message will be shown to user | |

|  |  |  |
| --- | --- | --- |
| Use Case | UC2: User modify post | |
| Actors | **User** | Operat |
| Pre-conditions | The user have logged in successfully | |
| Post-conditions | The post has been modified successfully | |
| Main success scenario | 1. User launch item management screen. 2. User press a particular post, and launch post detail. 3. User press “Edit” button. 4. User edit information of the post, including: Items and quantity. User can’t let one of mandatory fields empty). 5. User press “Save” button. 6. User is navigated back to Item management screen. 7. The post is updated and notification will push to all “Người Cho” who requested for updated items in the post. | |
| Alternative scenario 1 | 1. User let one of mandatory fields empty and press “Save”    1. an error message will be shown to user | |
| Alternative scenario 2 | 1. User delete an item which has requests.    1. an error message will be shown to user    2. User must handle all the requests for this item (Accept or Decline).    3. If user accept for one or more requests, the item’s status will be changed to : ”Đã cho”    4. If user decline all requests, user has 2 options: Move this item back to Store or delete this item. | |

|  |  |  |
| --- | --- | --- |
| Use Case | UC3: User delete post | |
| Actors | **User** | Operat |
| Pre-conditions | The user have logged in successfully | |
| Post-conditions | The post has been deleted successfully | |
| Main success scenario | 1. User launch item management screen. 2. User press a particular post, and launch post detail. 3. User press “Delete” button. 4. User is navigated back to Item management screen. 5. The post is deleted and notification will push to all “Người Cho” who have request to the post. | |
| Alternative scenario 1 | 1. User delete a post has one more items which has requests.    1. an error message will be shown to user    2. User must handle all the requests for this item (Accept or Decline).    3. If user accept for one or more requests, the item’s status will be changed to : ”Đã cho”    4. If user decline all requests, user has 2 options: Move this item back to Store or delete this item. | |

# Non-functional Requirements

## Operating System Support

Scientific Workflows App supports iOS 11 and above.

# ANALYZE AND DESIGN

## Activity Diagram

## Class Diagram

## Database

…

# APPENDIX A: [TBD]