- Use Case Diagram

- Use Case Descriptions

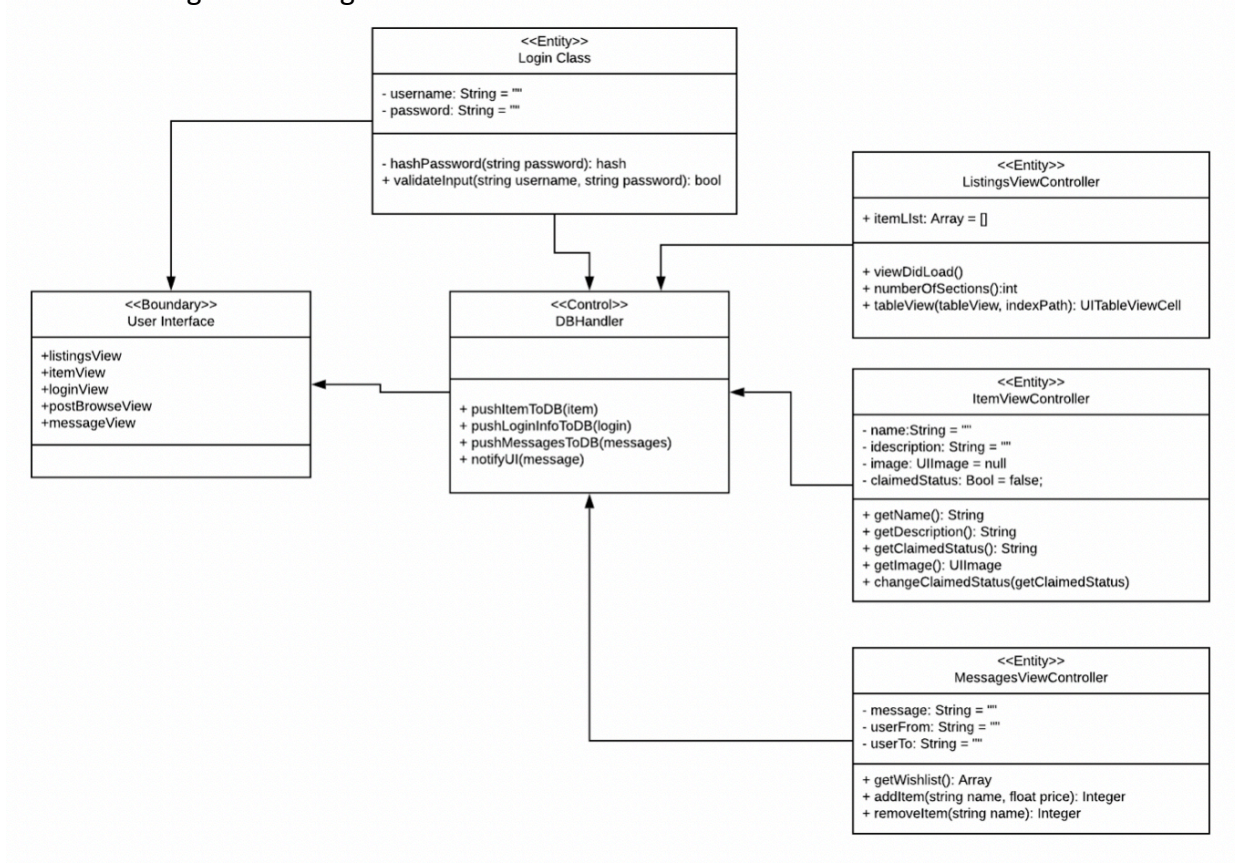|	
Use Case: Claim Item	
Primary Actors: Item Claimer	
Goal in Context: To claim an item that has been posted.	
Precondition: System fully configured, and actor is logged into their account.	
Trigger: The actor wants to claim an item that was posted.	

Scenario:
    1. Actor clicks browse items button on home screen.
    2. Actor taps on item they want from listings.
    3. Actor taps the claim item button on the item's page.
    4. Actor is prompted to message the user who posted.	

Exceptions: 1. Item has already been claimed or otherwise unavailable.	
Priority: High. Vital to project.	
When Available: First increment.	
Frequency of Use: Often.	
Channel to Actor: iOS device and internet connection.	
Secondary Actors: None.	
Open Issues: Creating own API and dealing with database to make this work.	

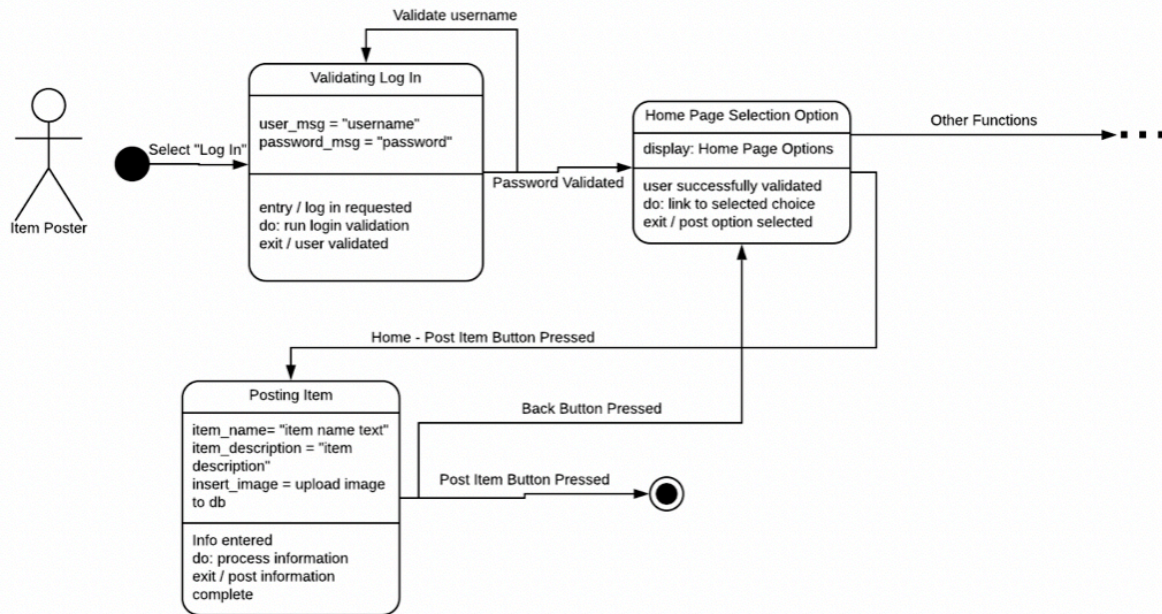| |
|---|
| **Use Case:** Claim Item |
| **Primary Actors:** Item Claimer |
| **Goal in Context:** To claim an item that has been posted. |
| **Precondition:** System fully configured, and actor is logged into their account. |
| **Trigger:** The actor wants to claim an item that was posted. |
| **Scenario:**
　1. Actor clicks browse items button on home screen.
　2. Actor taps on item they want from listings.
　3. Actor taps the claim item button on the item's page.
　4. Actor is prompted to message the user who posted. |
| **Exceptions:**
　1. Item has already been claimed or otherwise unavailable. |
| **Priority:** High. Vital to project. |
| **When Available:** First increment. |
| **Frequency of Use:** Often. |
| **Channel to Actor:** iOS device and internet connection. |
| **Secondary Actors:** None. |
| **Open Issues:** Creating own API and dealing with database to make this work. |

Use Case: Post Item with Photo and Description
Primary Actors: Item Poster
Goal in Context: Create and upload a post with a photo and description of an item being given away.
Precondition: System fully configured, and actor is logged into their account.
Trigger: The actor wants to post a new item to be given away.
Scenario: 　1. Actor clicks the post item button on the home screen. 　2. Enter item name. 　3. Insert picture. 　4. Enter description. 　5. Press post button.
Exceptions: 　1. Invalid picture. 　2. Invalid item name. 　3. Invalid item description.
Priority: High. Vital to project.
When Available: First increment.
Frequency of Use: Often.
Channel to Actor: iOS device and internet connection.
Secondary Actors: None.
Open Issues: Creating own API and dealing with database to make this work.

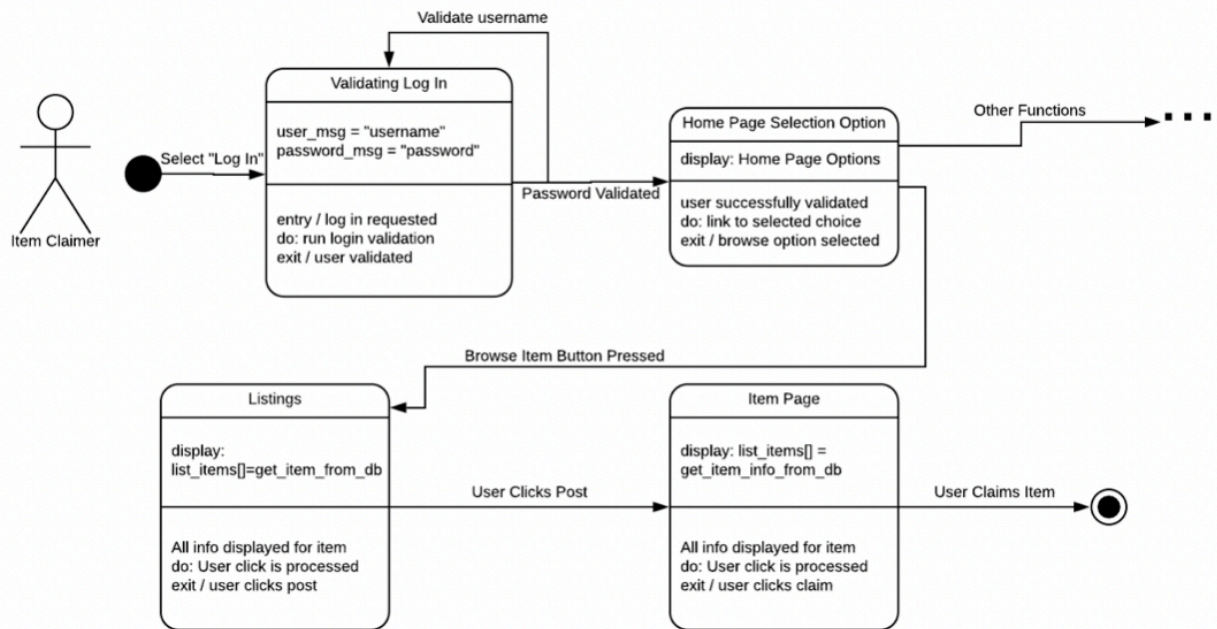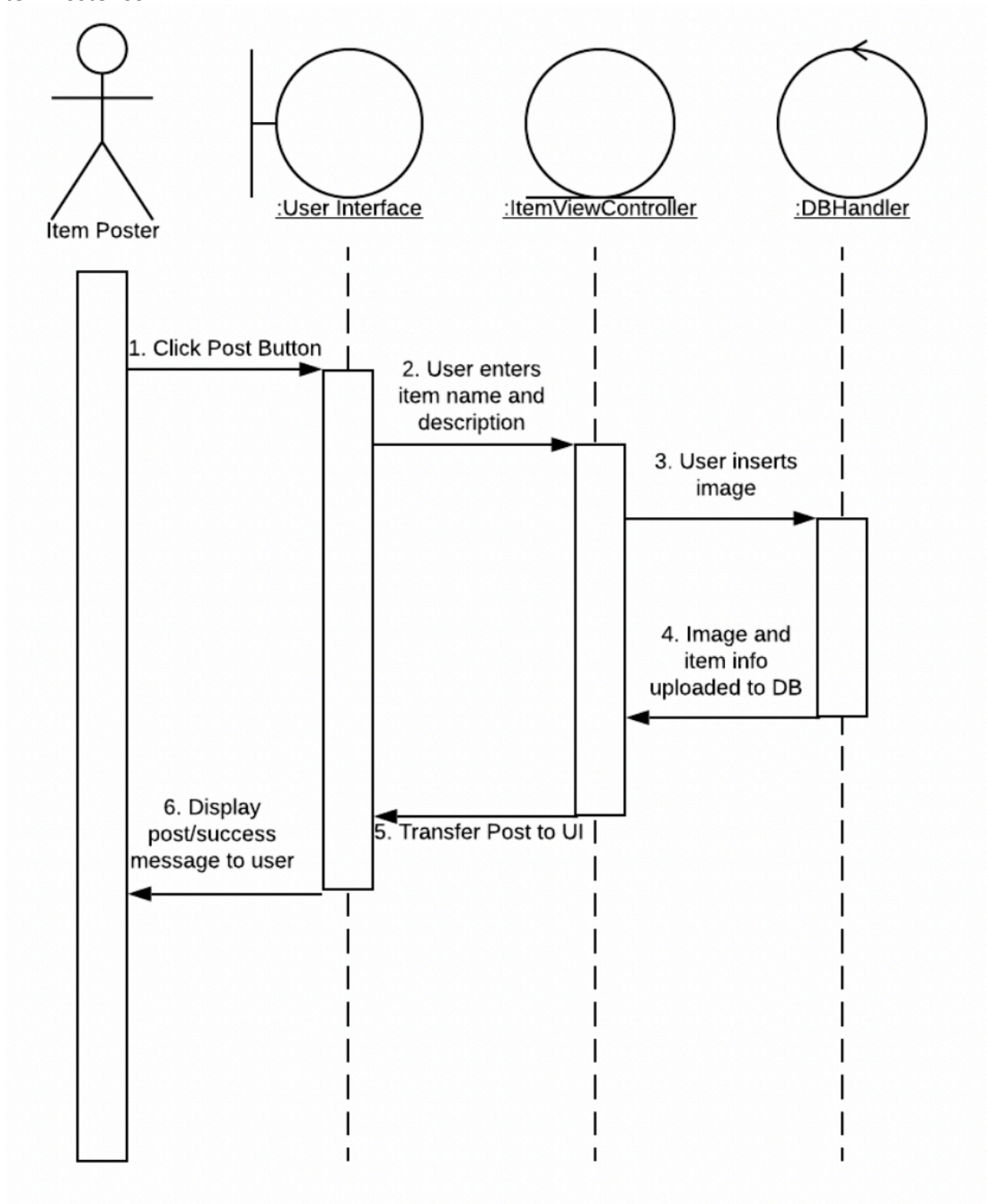| |
|---|
| **Use Case:** Message Users |
| **Primary Actors:** Item Poster, Item Claimer |
| **Goal in Context:** Message a user after claiming an item or messaged a user who contacted you about your posted item. |
| **Precondition:** System fully configured, actor is logged into their account, and has just claimed a posted item or has a received a message about their posted item. |
| **Trigger:** The actor has claimed an item and wanted to message for <u>details</u> or the actor wants to reply to a message they received about their posted item. |
| **Scenario:**
    1. Press the messages button after claiming an item or receiving a message first.
    2. Enter the message into the text field.
    3. Press the send message button. |
| **Exceptions:**
    1. Message contains non-allowed words. |
| **Priority:** Medium-High. Important to project but could be worked around if not working. |
| **When Available:** Planned first increment, could be second though. |
| **Frequency of Use:** Often. |
| **Channel to Actor:** iOS device and internet connection. |
| **Secondary Actors:** None. |
| **Open Issues:** Dealing with chat database and displaying message history between all users. |

- Detailed Deisgn Class Diagram

- Item Poster Statechart

Validate username

Validating Log In

user_msg = "username"
password_msg = "password"

entry / log in requested
do: run login validation
exit / user validated

Select "Log In"

Item Poster

Password Validated

Home Page Selection Option

display: Home Page Options

user successfully validated
do: link to selected choice
exit / post option selected

Other Functions

Home - Post Item Button Pressed

Posting Item

item_name= "item name text"
item_description = "item description"
insert_image = upload image to db

Info entered
do: process information
exit / post information complete

Back Button Pressed

Post Item Button Pressed

- Item Claimer Statechart

Validate username

Validating Log In

user_msg = "username"
password_msg = "password"

entry / log in requested
do: run login validation
exit / user validated

Select "Log In"

Item Claimer

Password Validated

Home Page Selection Option

display: Home Page Options

user successfully validated
do: link to selected choice
exit / browse option selected

Other Functions

Browse Item Button Pressed

Listings

display:
list_items[]=get_item_from_db

All info displayed for item
do: User click is processed
exit / user clicks post

User Clicks Post

Item Page

display: list_items[] = get_item_info_from_db

All info displayed for item
do: User click is processed
exit / user clicks claim

User Claims Item

- Item Poster SSD

Item Poster

:User Interface

:ItemViewController

:DBHandler

1. Click Post Button

2. User enters item name and description

3. User inserts image

4. Image and item info uploaded to DB

5. Transfer Post to UI

6. Display post/success message to user

- Item Claimer SSD