Iteration 4 - Students Requirements Proposal

Team ID/Name: __SA1__

TA Name: _Yi Wang_

Team Application Theme	**Approved**
Reports (incidents, user information, status detail, resources)	Y

Student 1	**Proposal**	**Approved**
James	**Step 1: High-Level Requirement** Use Case Name: Report Detailed Emergency Status Use Case Brief Description: In this use case, users are able to share the details of their status to let others know what specific emergency situation the user is in. So this is the detailed version of the previous share status use case. (Users are able to specify their location description, picture of her/his current environment, and more survey questions)	Y
	Faculty/TA Comments:	
	Step 2: Clickable UI Mockups Link: https://drive.google.com/open?id=11hIUz_ZmYlzgzCfk3XfE6AOQWyKyE6Er	Y
	Faculty/TA Comments:	
	Step 3: Elaboration - Use Case Specification See below (min 6 steps in basic flows and 6 alternative flows or rules)	Y
	Faculty/TA Comments:	
	Step 4: Use-Case Analysis Model (OOA) See below	Y
	Faculty/TA Comments:	

Use Case Specification

Participating Actors

The use case is initiated by a Citizen.

Brief Description

In this use case, users are able to share the details of their emergency status in order to let others know what specific emergency situation the user is in.

Assumption

The Citizen is logged into the system and selects Emergency status in the share status popup window. The available status is listed in the **User Status rule** below.

Flow of Events

Basic Flow

1. The use case starts when the citizen clicks the save button.
2. The system closes the popup window and switches to the status detail preview page which is under the status tab of the header and shows a preview of the emergency details that was added before. There are three kinds of emergency status details defined under the **emergency status detail type rule** below
3. The user clicks the edit button.
4. The system replaces the status detail preview with a general status detail edit page. The general edit page shows all the previously entered detail information of the emergency status listed in the **emergency status detail type rule** below. Texts are in editable textarea and pictures have a reupload and delete button.
5. The user elects to edit the textarea under the “Briefly describe your situation” subtitle.
6. The user scrolls down to the bottom of the screen and clicks the save button.
7. The system saves the changes and returns to the status detail preview page and the use case ends.

Alternative Flows

- A1. In step 7 of basic flow, if the user elects to share her location then the user can type description in the textarea under “Share your location ” for detailed information with words. The flow returns to step 8 of basic flow.
- A2. In step 7 of basic flow, if the user clicks the “Add...” button under the add pictures and descriptions subtitle. The system prompts the user to choose a picture from her device or take a picture. After the user chooses a picture or takes a picture, she clicks on the confirm button. Then the system shows a popup screen showing the uploaded picture. The user can type in the textarea below the picture to give the picture a description. The user can also click on the reupload button to reupload a new picture. The user clicks save and the system returns to the general edit page and shows the newly uploaded picture at the bottom of the “add pictures and descriptions” section. The flow returns to step 8 of basic flow.

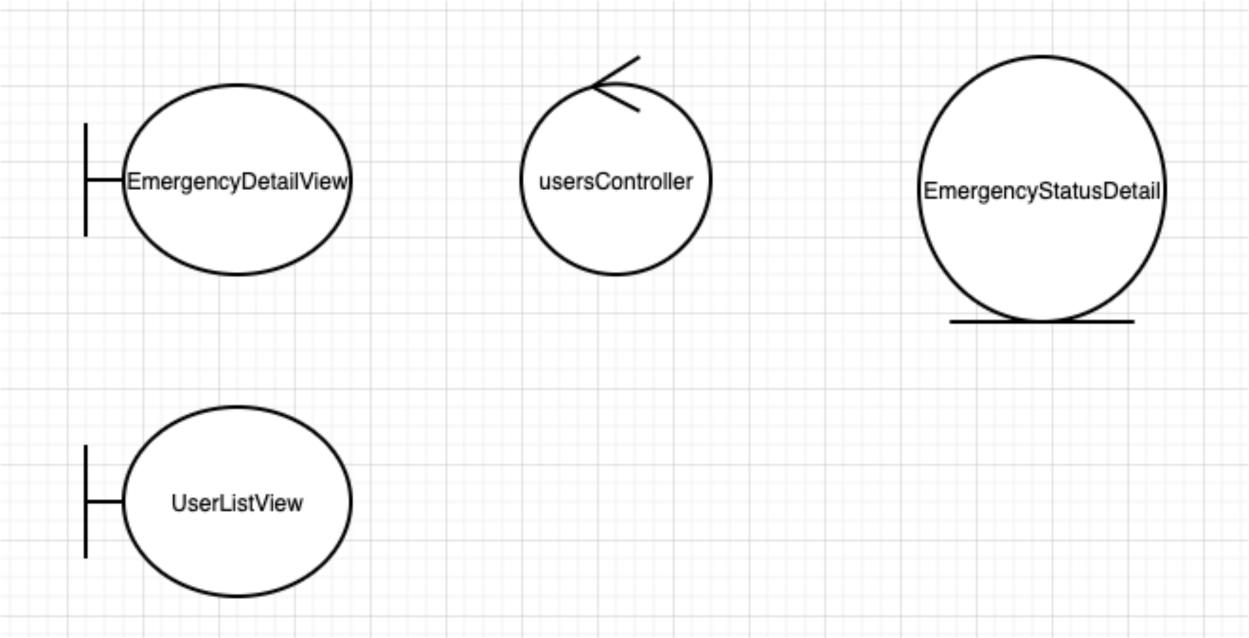
- A3. In step A3 of Alternative Flows, if the user changes her mind and doesn’t want to upload a picture, she can click the cancel button when she is choosing a picture. The flow returns to step 8 of basic flow.
- A4. In step A3 of Alternative Flows, if the user changes her mind after choosing a photo or taking a photo to upload, she clicks the cancel button in the popup screen. The flow returns to step 8 of basic flow.
- A5. In step 7 of basic flow, if the user clicks the “Reupload...” button below one of the pictures in the “add pictures and descriptions” section. The system prompts the user to choose a picture or take a photo. The user selects a picture and the system returns to the general edit page. The flow returns to step 8 of basic flow.
- A6. In step 8 of basic flow, if the user changes her mind and doesn't want to edit anything or she wants to discard all the changes. She scrolls down to the bottom of the page and clicks on the cancel button. The flow returns to step 9 of basic flow.
- A7. In step 9 of basic flow, after saving the emergency detail, if the user wants to see how her own emergency status detail looks to other users or wants to see another user’s emergency status, the user clicks on the Users button in the application header, then the system shows the users list. After that, the user clicks on the emergency icon of a user (either her own or other users’), then the system generates a popup screen that shows the emergency status detail of that user. In the end, the user closes the popup window and the use case ends.

Rules

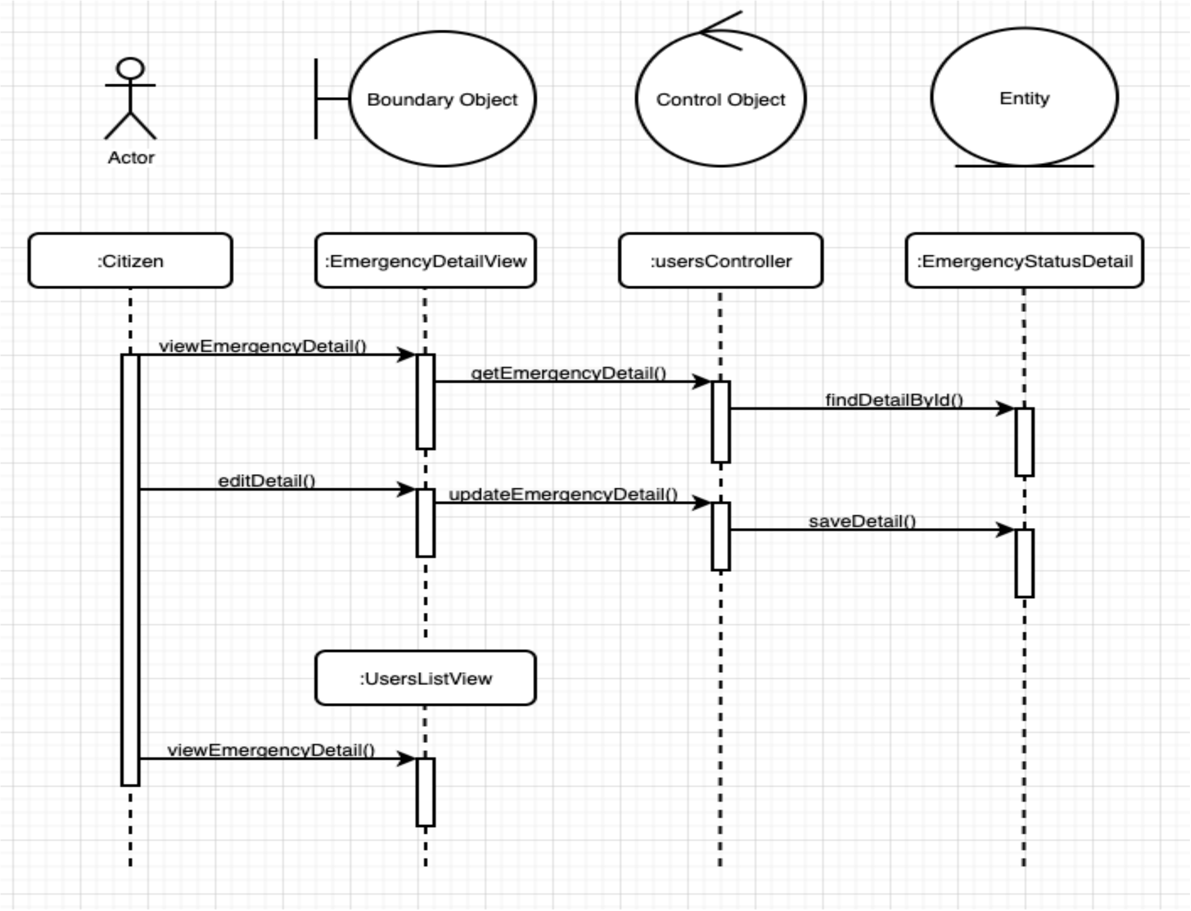
- **User status**
    - OK
    - HELP
    - EMERGENCY
    - UNDEFINED.
- **Emergency status detail type**
    - A brief description of the situation
    - Share location
    - Share pictures with or without description

Use Case Analysis Model (OOA)

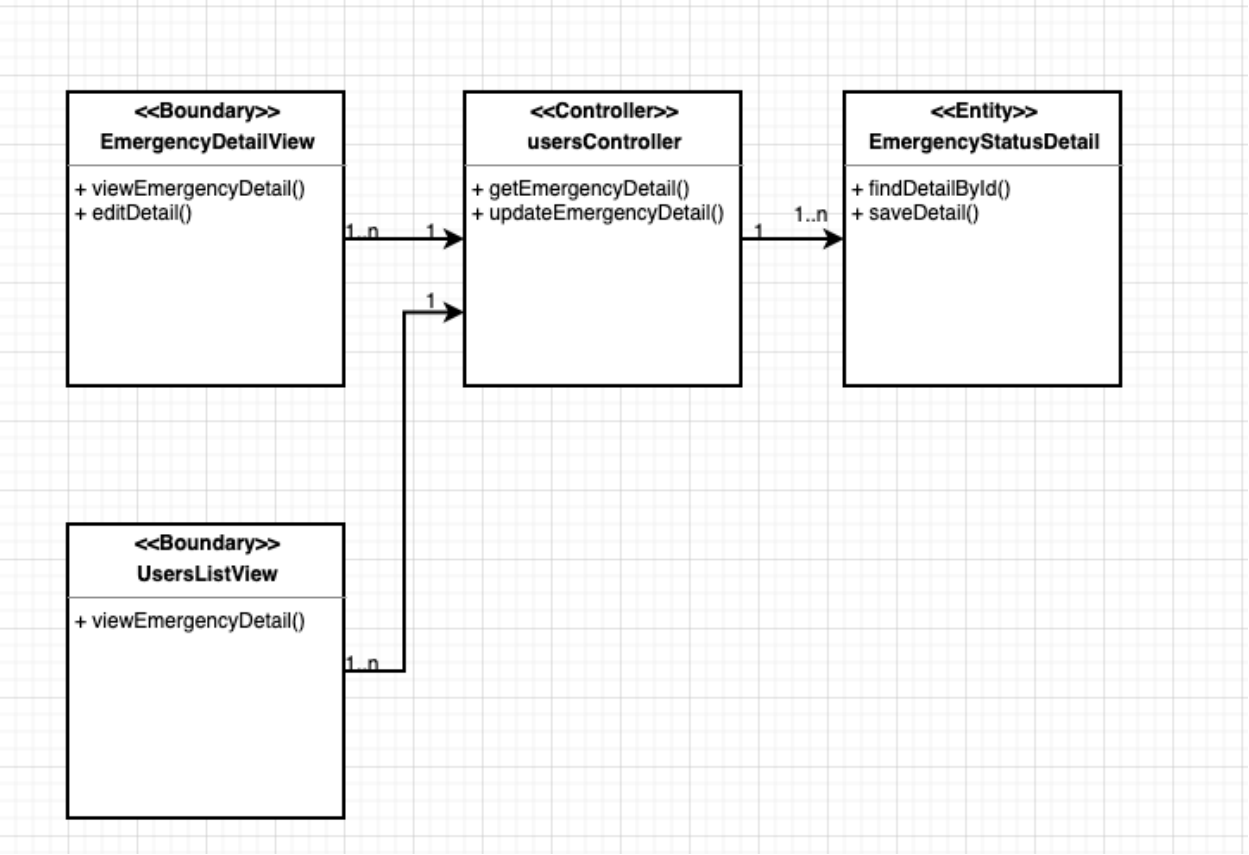
Analysis classes:



Sequence Diagram:


Actor
Boundary Object
Control Object
Entity
:Citizen
:EmergencyDetailView
:usersController
:EmergencyStatusDetail
viewEmergencyDetail()
getEmergencyDetail()
findDetailById()
editDetail()
updateEmergencyDetail()
saveDetail()
:UsersListView
viewEmergencyDetail()

Class Diagram:



Mapping Between Analysis Classes and Code:

Analysis Classes	**Implementation Elements** (e.g. modules, files, components, databases)
EmergencyDetailView	emergency_status_detail.ejs
UsersListView	user_list.ejs
EmergencyDetailController	EmergencyStatusDetailController.js
EmergencyStatusDetail	statusDetail.js

Quality Report (for your individual branch only, not master)

Unit tests you added in Iteration 4

Module/File Tested	**# Tests**	**# Failing Tests**

emergencyStatusDetail.js	8	0

Integration tests you added in Iteration 4

Module/File Tested	# Tests	# Failing Tests
emergencyStatusDetail.js	8	0

Bugs found through testing in Iteration 4

Module/File	Description	Fixed? (Y/N/Partially)

Coverage (your branch only)

Line Coverage	84.26%
Branch coverage	79.34%

All files

84.26% Statements `637/756` **79.34%** Branches `96/121` **73.36%** Functions `179/244` **84.26%** Lines `637/756`

Press *n* or *j* to go to the next uncovered block, *b*, *p* or *k* for the previous block.

File ▲		Statements		Branches		Functions		Lines	
repo		81.08%	60/74	28.57%	4/14	25%	1/4	81.08%	60/74
repo/controllers		83.85%	218/260	86.54%	45/52	69.88%	58/83	83.85%	218/260
repo/middleware		82.14%	23/28	75%	3/4	85.71%	6/7	82.14%	23/28
repo/model		82.27%	246/299	86.27%	44/51	77.93%	113/145	82.27%	246/299
repo/routes		94.32%	83/88	100%	0/0	0%	0/4	94.32%	83/88
repo/utils		100%	7/7	100%	0/0	100%	1/1	100%	7/7