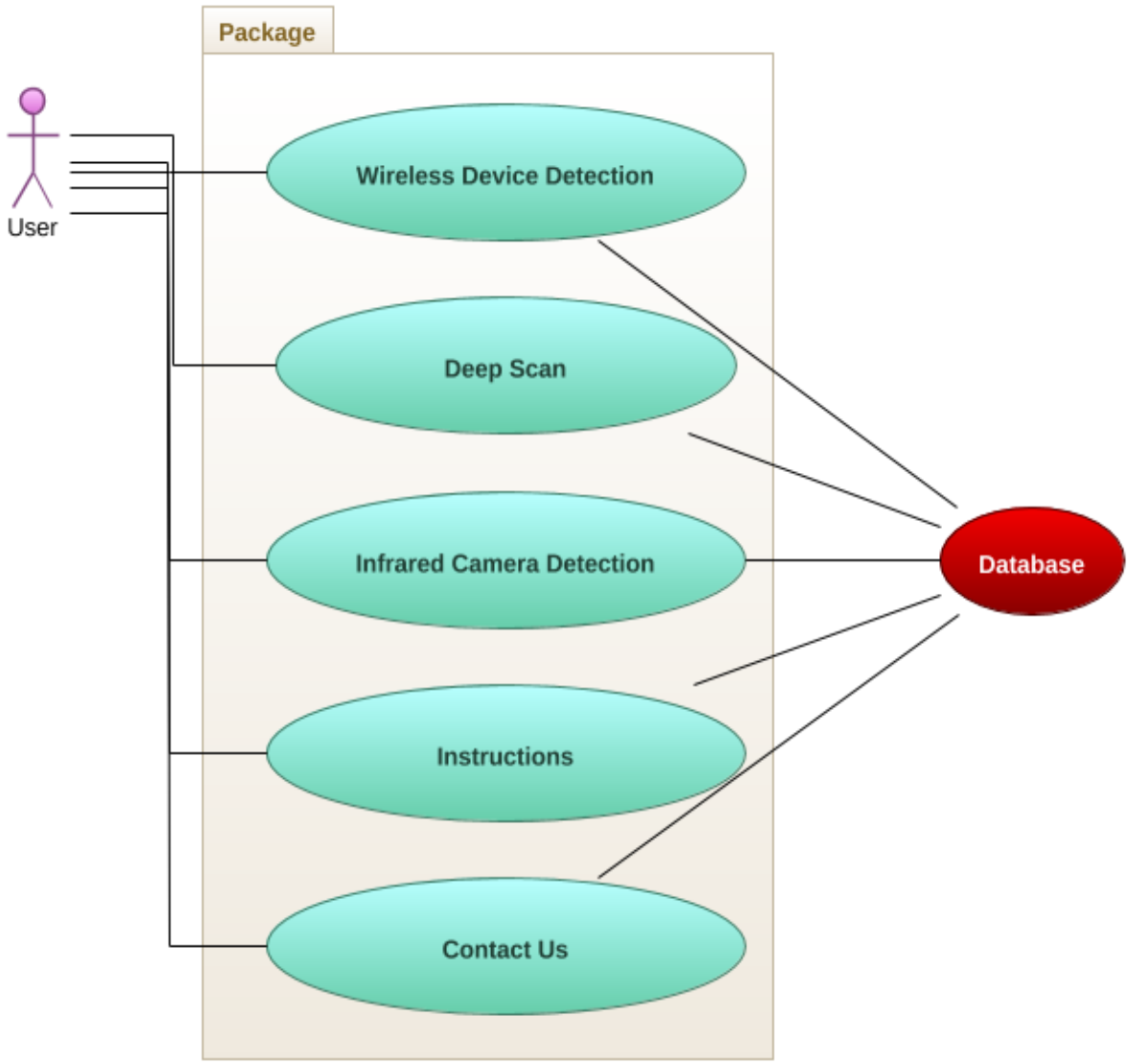
Angelique Refahiyat
UML Diagram – ISIT 322
04-25-2020
Prof: Mr. Phil Duncan

DeviceDetect is designed to detect wireless devices available up to one mile from its users' location including wireless devices and infrared cameras. The simple DeviceDetect functionality among other available apps on the market makes it to be unique in terms of its fast installation and its responsiveness.
DeviceDetect wireless device detector app, uses the most recent technology and high-quality digital sensors to give its user the most expected result. The highest capability of the installed sensor in DeviceDetect app allows its users to accelerate the detection's process many times faster than the other wireless detection apps.

Scope:

The simple installation of DeviceDetect allows its users to remotely access the features of this app everywhere they go regardless of their locations. The detection sensor installed on DeviceDetect processes fast as users' network internet connection. The faster internet connection will produce the faster and more accurate detection result. DeviceDetect will trace its users' activities using a non-relational database for their further requests and records. DeviceDetect functionality is limited to detect all wireless devices and infrared cameras to provide a better and specific result to its users. In terms of DeviceDetect compatibility, the app is compatible for all IOS users to take advantage of its features regardless of the version of their devices. DeviceDetect free installation makes the app affordable for its users to be members of DeviceDetect family at no cost. The security measurement implemented on DeviceDetect, allows its users to keep their provided information safe and secure from not being used or stolen by any third-party advertisement agencies or any other illegal types of usages.

Use Case Diagram:



List of Steps:

1 Establish Internet Wi-Fi Connection locally
- 1.2 Device compatibility to connect to the internet
- 1.3 Verify modem is connected to a wireless router, or a wireless gateway properly
- 1.4 Establish well Wi-Fi network connection

2 Launch the app
- 2.1 Connect to remote server using an appropriate network security tool
- 2.2 Interface
  - 2.2.1 Design Layout
    - 2.2.1.1 Launch Screen
    - 2.2.1.2 Main Menu
    - 2.2.1.3 Screen that displays all the sub-menu[s]
    - 2.2.1.4 Post processing screen
      - 2.2.1.3 Ensure app supports remote notification
      - 2.2.1.4 minification [by Pleease]
  - 2.2.2 Design Splash screen
    - 2.2.2.1 Initial Launch storyboard
    - 2.2.2.2 Confirm user immediate feedback that the app has started and loaded
    - 2.2.2.3 Level up app responsiveness
    - 2.2.2.4 Display branding information to user
    - 2.2.2.5 Short and timewise
  - 2.2.3 Image-Logo type Splash screen sequence
    - 2.2.3.1 Branded screen
    - 2.2.3.2 UI placeholder, UIKit
    - 2.2.3.3 UIApplication.LaunchOptionsKey
    - 2.2.3.4 Passed along a launch options dictionary to app
    - 2.2.3.5 The app itself

3 Database Connection
- 3.1 Precondition
  - 3.1.1 Admin credential validity check to connect
  - 3.1.2 App interacts properly with a designated database collection
- 3.2 Supportive client's IOS platform
- 3.3 Access and transfer data securely
  - 3.3.1 data at rest
  - 3.3.2 data in motion
- 3.4 Perform a flexible data model
  - 3.4.1 articulate the model requirements efficiently
- 3.5 Resolve data conflict
  - 3.5.1 Using three-way merge approach
- 3.6 Sync at the right time
  - 3.6.1 Using conditional replication approach
    - 3.6.1.1 To replicate data only under certain conditions

3.6.1.1.1 When the device is on Wi-Fi
3.6.1.1.2 When the device has enough battery power
3.7 Sync with the right partition[s]
3.7.1 Using configurable [star] sync topology support
3.7.1.1 Configure the app to operate offline
4 Login
4.1 User
4.1.1 Registered User
4.1.1.1 Provide Username
4.1.1.2 Provide Password
4.1.2 Non-Registered User, Create account
4.1.2.1 Provide a valid email
4.1.2.2 Provide a valid password
4.1.2.3 Verify entered password
4.1.2.4 Verify email
4.1.2.5 Login
5 Select Menu
5.A Precondition
5.A.1 Select Wireless Device Detection
5.A.1.1 Display the sub-menu
5.A.2 Wireless Device Detection functionality
5.A.2.1 Detect wireless devices process
5.A.3 Post Wireless Device Detection functionality
5.A.3.1 Store devices found in database
5.A.4 End of session
5.A.4.1 Keep the record in database
5.B Precondition
5.B.1 Select Deep Scan
5.B.1.1 Display the sub-menu
5.B.2 Wireless Device Detection functionality
5.B.2.1 Detect selected wireless device using device camera
5.B.3 Post Deep Scan functionality
5.B.3.1 Store device found in database
5.B.4 End of session
5.B.4.1 Keep the record in database
5.C Precondition
5.C.1 Select Infrared Camera Detection
5.C.1.1 Display the sub-menu
5.C.2 Infrared Camera Detection functionality
5.C.2.1 Detect infrared cameras process
5.C.3 Post Infrared Camera Detection functionality
5.C.3.1 Store infrared cameras found in database
5.C.4 End of session
5.C.4.1 Keep the record in database

5.D Precondition

5.D.1 Select Instructions

5.D.1.1 Display the sub-menu

5.D.2 Wireless Instructions functionality

5.D.2.1 Step through user guide process

5.D.3 Post Instructions functionality

5.D.3.1 Ask user to save a copy

5.C.3.2 Store an instructions' copy into user device if selected

5.D.4 End of session

5.D.4.1 Store an instructions' copy into user device if selected

5.D.4.2 Store an instructions' copy in database if selected

5.E Precondition

5.E.1 Select Contact Us

5.E.1.1 Display the sub-menu

5.E.2 Contact Us functionality

5.E.2.1 Contact form process

5.E.2.2 User to complete a questionnaire

5.E.2.3 Submit the questionnaire

5.E.3 Post Contact Us functionality

5.E.3.1 Send the questionnaire's copy to admin

5.E.3.2 Send the questionnaire's copy to user

5.E.4 End of session

5.E.4.1 Store submitted confirmation success message in database

6 Cancel

6.1 Precondition

6.1.1 Select Cancel

6.1.2 Display an alert asking user to logout or more options

6.2 Cancel functionality

6.2.1 process user request

6.2.1.1 Log out

6.2.1.2 Log out process

6.2.2.1 Display More Options

6.2.2.2 More Options process

6.3 Post Cancel functionality

6.3.1 Send user request to the remote server

6.4 End of session

6.4.1 In case of log out

6.4.1.1 Delete user data from the local storage

6.4.1.2 Disconnect user from the remote server

6.4.2 In case of More Options

6.4.2.1 More Options process

7 More Options

7.1 Precondition

7.1.1 Select More Options
7.1.1.1 Display More Options
7.2 More Options functionality
7.2.1 Main Menu
7.2.1.1 Select Main Menu
7.2.1.1.2 Main Menu process
7.2.2 Cancel
7.2.2.1 Select Cancel
7.2.2.2 Cancel process
7.2.3 Contact Us
7.2.3.1 Select Contact Us
7.2.3.2 Contact Us process
7.3 Post More Options functionality
7.3.1 Select Main Menu
7.3.1.1 Post Main Menu process
7.3.2 Select Cancel
7.3.2.1 Post Cancel process
7.3.3 Select More Options
7.3.3.1 Post More Options process
7.4 End of session
7.4.1 Select Main Menu
7.4.1.1 End of session of Main Menu process
7.4.2 Select Cancel
7.4.2.1 End of session of Cancel process
7.4.3 Select More Options
7.4.3.1 End of session of More Options proces
8 Log out
8.1 Precondition
8.1.1 Select Log out
8.1.1.2 Display Log out
8.2 Log out functionality
8.2.1 User request will send to the remote server
8.2.2 The remote server will process the user request
8.2.3 The remote server, automatically will disconnect user from the system
8.3 Post Log out functionality
8.3.1 Send an alert notification to user
8.3.2 System deletes the token from user device local storage
8.4 End of session
8.4.1 User disconnect
8.4.2 User exit

Process requirements

following are among the inherent requirements that the wireless devices' detection app system must be able to handle.

9.1 Online database

The system must be able to send, receive and trigger user requests and activities to the online database system.

9.2 Data integrity

Commit user requests that are completed and/or rollback unfinished or time-out requests.

9.3 Data validation

Data error from the user's end and from the back-end database-processing end must be gracefully handled. There will be data validation and error-handling routines as part of the online user login and activities procedure.

9.4 Performance

Must resolve locking and hanging issues and handle concurrent use of the system on an online - offline basis. Send, receive, and display user messages to assist the over-all user experience.

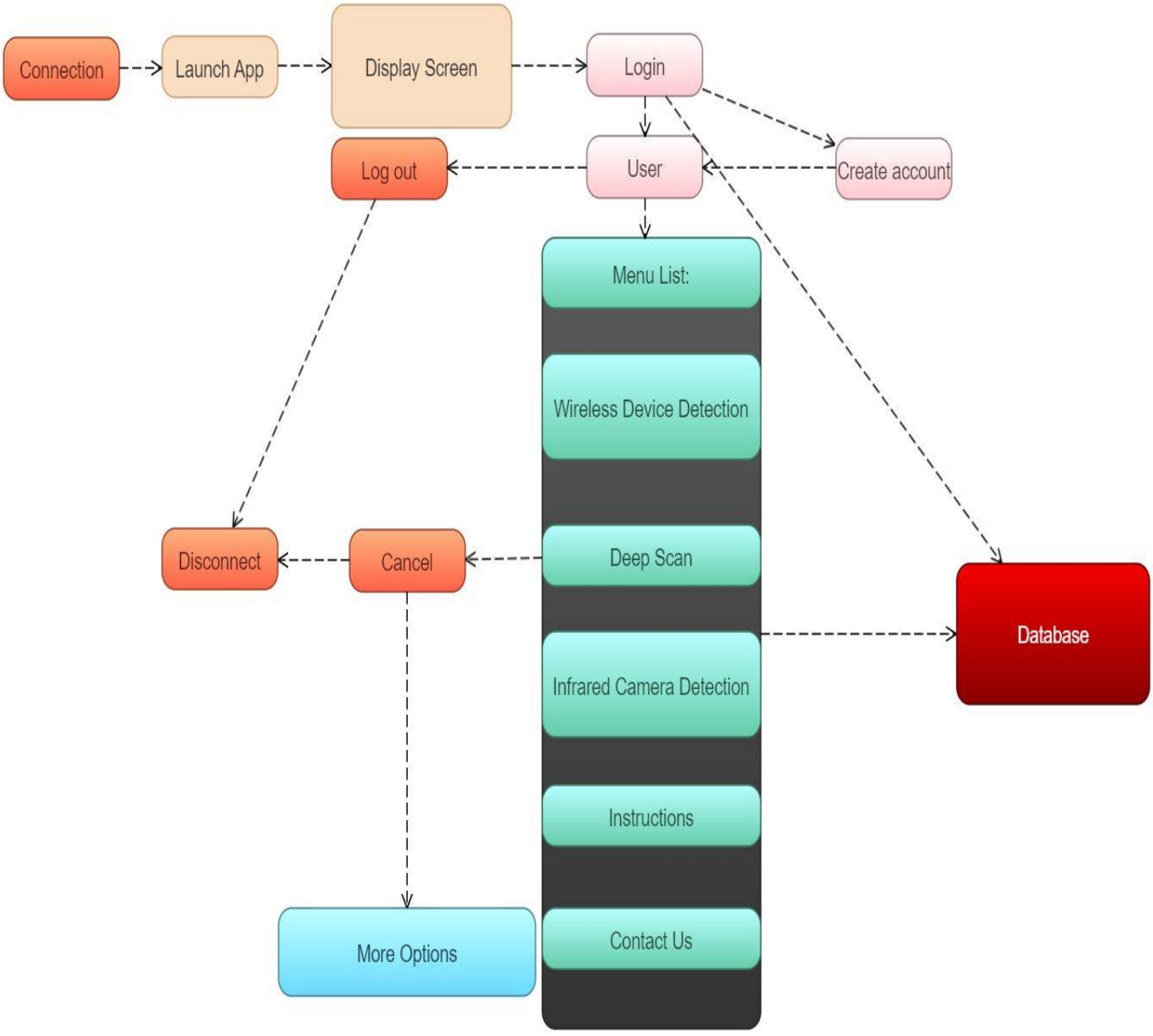
9.5 Data repository

The wireless devices' detection app system will maintain the existing online database as the main repository of data

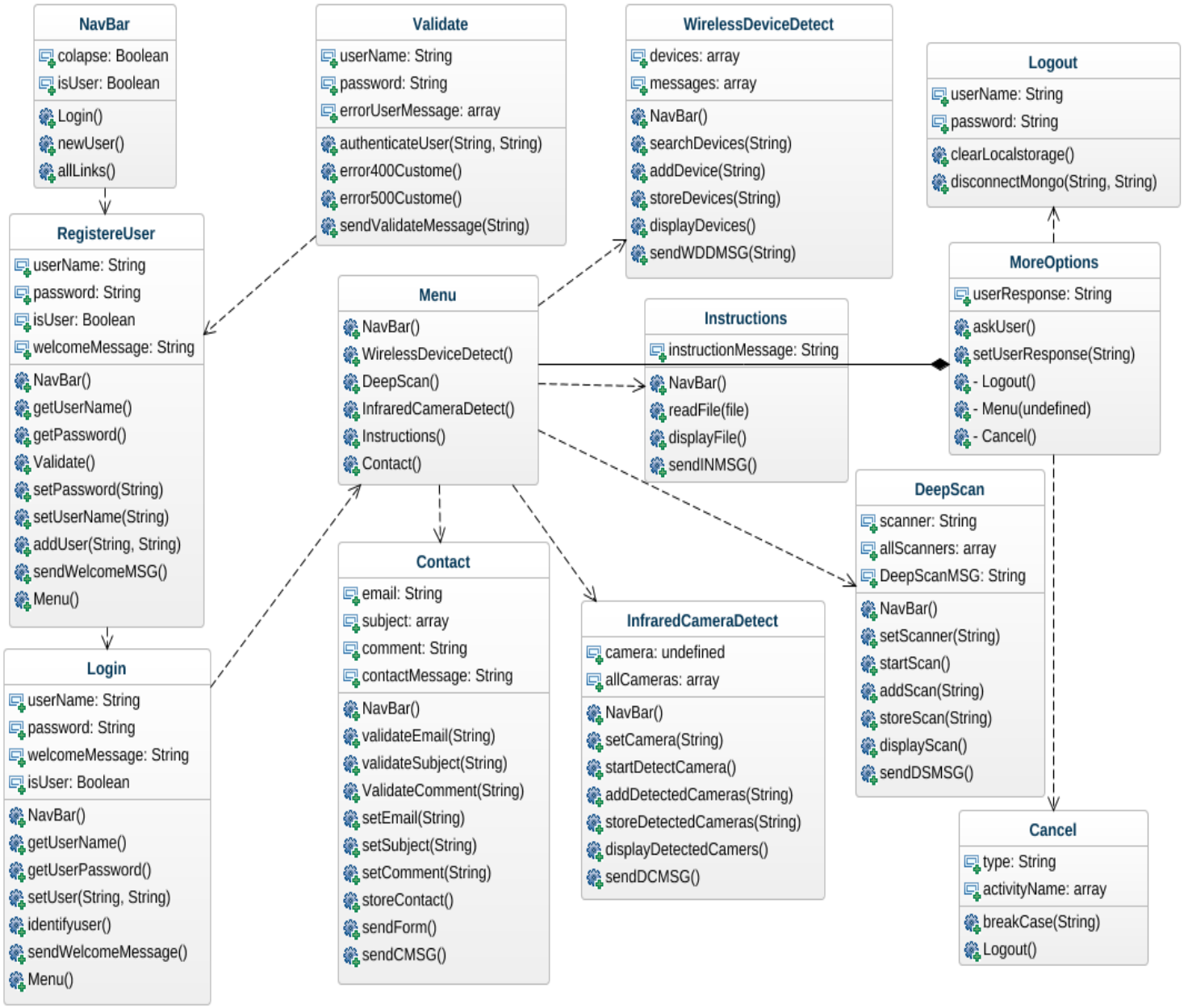
Test Main Steps [by its priority and importance]

id	Step	id	Test
1	Establish Internet Wi-Fi Connection	1	User connected to the local server with no glitches or firewall problem with an appropriate speed.
2	Launch DeviceDetect	2	App's current version is compatible with the user IOS device & downloadable & all the app's contents display on user's device are readable and executable.
3	Database connection security check	3	User's credential verifiable, with no breakage, errors, accidently shutdowns.
4	Login	4	System permits user to access credentials as its recognized & allow the new users to create a secured account to login with no rejection or unknown errors.
5	Menu	5	Sub-menu contents are executable and accessible.
6	Cancel	6	Upon user request, the running operation cancels out, and lead the user in a right direction to follow.
7	More Options	7	Upon user request, the More Options menu works appropriately and allows user to choose a desired-selected path.
8	Log out	8	Upon user request to log out, system functions as it designed in steps provided to disconnect user and let user to securely exit.

Activity Diagram



Class Diagram:



Sequence Diagram: