**Software Design Specification**

**1. Introduction**

The Software Design Document is a document to provide documentation which will be used to aid in software development by providing the details for how the software should be built. Within the Software Design Document are narrative and graphical documentation of the software design for the project including use case models, sequence diagrams, object behaviour models, and other supporting requirement information.

**1.1 Purpose of this document**

This document will define the design of the one runway simulator. It contains specific information about the expected input, output, classes, and functions. The interaction between the classes to meet the desired requirements are outlined in detailed figures at the end of the document

**1.2 Scope of the development project**

This application is for the Library of NIIT University which will be linked to the upcoming website of NU-LIRC. This will enable the students, faculty and staff to access the benefits of the NU-LIRC website on an Android application. The scope of this product does not include iOS and windows phones.

**1.3 Definitions, acronyms, and abbreviations**

IEEE: Institute of Electrical and Electronics Engineers

iOS: iPhone Operation System

NU-LIRC: NIIT University Library and Information Resource Centre

SDS: Software Design Specification

**1.4 References**

* IEEE SDS template
* UML version 2.0
* NU-LIRC website
* NIIT Library Website
* Firebase Database

**1.5 Overview of document**

This SDS is divided into seven sections with various sub-sections. The sections of the Software Design Document are:

**1. Introduction:** describes about the document, purpose, scope of development project definitions and abbreviations used in the document.

**2. Conceptual Architecture/Architecture Diagram:** describes the overview of components, modules, structure and relationships and user interface issues.

**3. Logical Architecture:** describes Logical Architecture Description and Components.

**4. Execution Architecture:** defines the runtime environment, processes, deployment view.

**5. Design Decisions and Trade-offs:** describes the decisions taken along with the reason as to why they were chosen over other alternatives.

**6. Pseudocode for components:** describes pseudocode, as the name indicates.

**7. Appendices:** describes subsidiary matter if any.

**2. Conceptual Architecture/Architecture Diagram**

NA

**2.1 Overview of modules / components**

NA

**2.2 Structure and relationships**

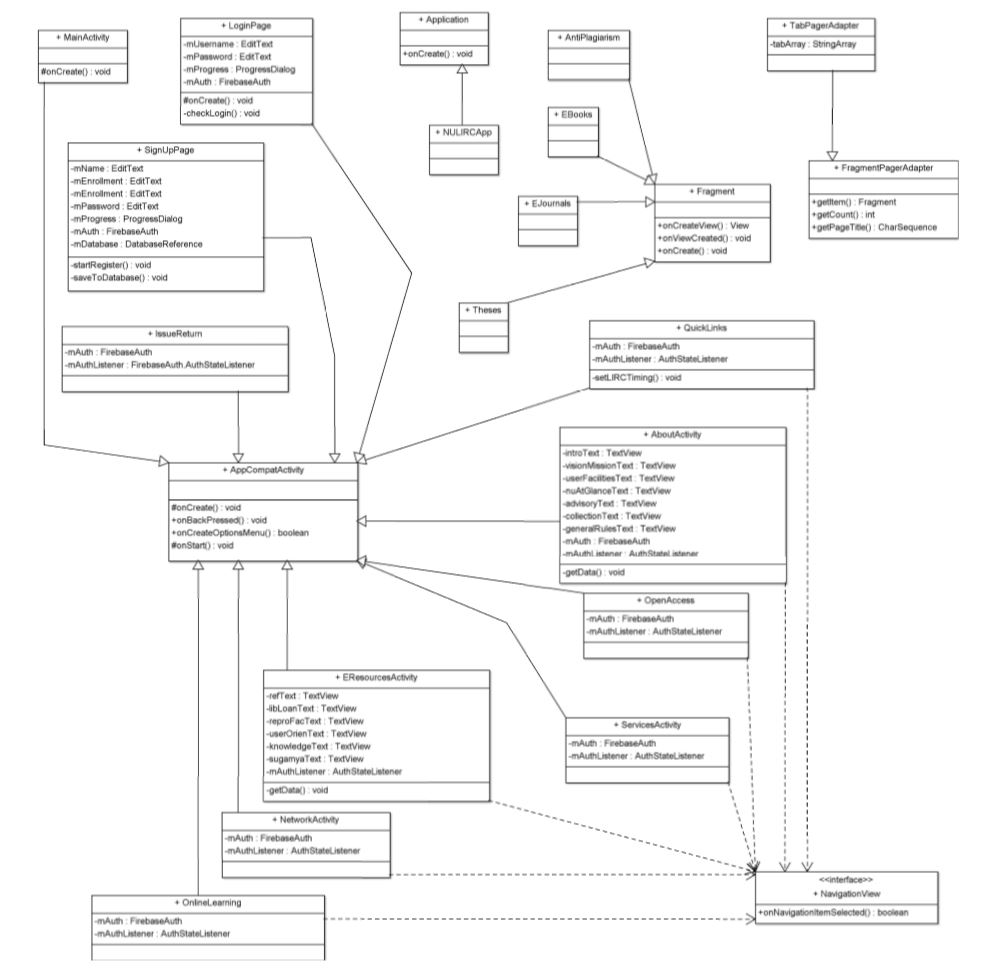
NA

**2.3 User interface issues**

NA

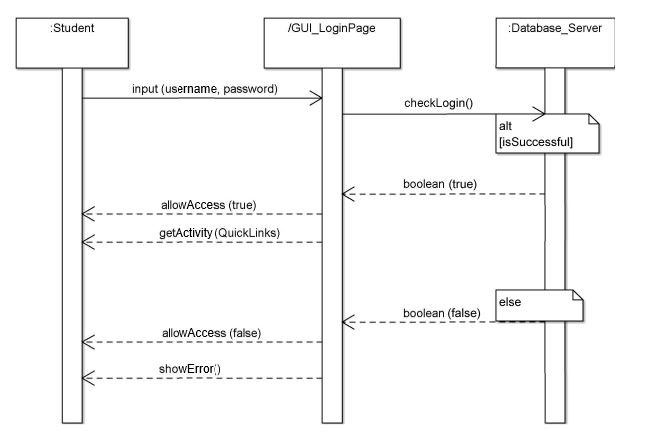
**3. Logical Architecture (Class Diagram, Sequence Diagram, State Diagram)**

**Class Diagram**

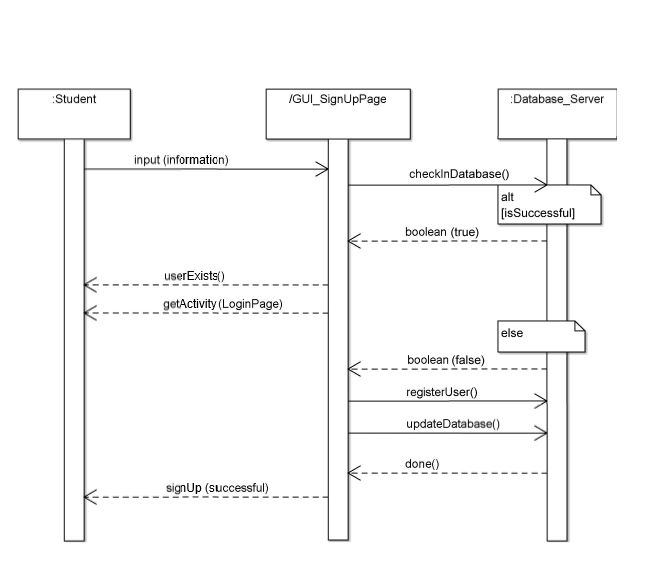


**Sequence Diagrams:**

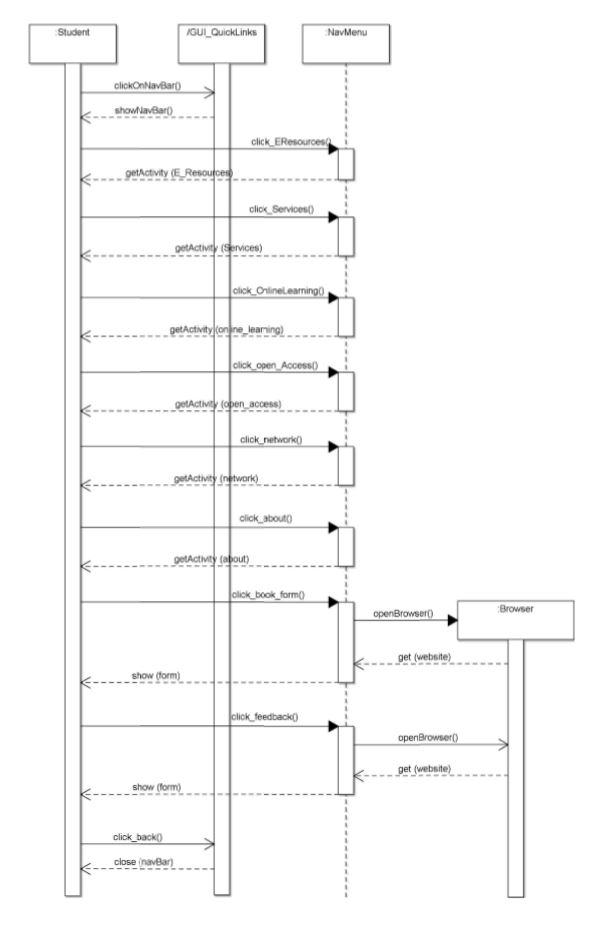
**LOGIN PAGE**

****

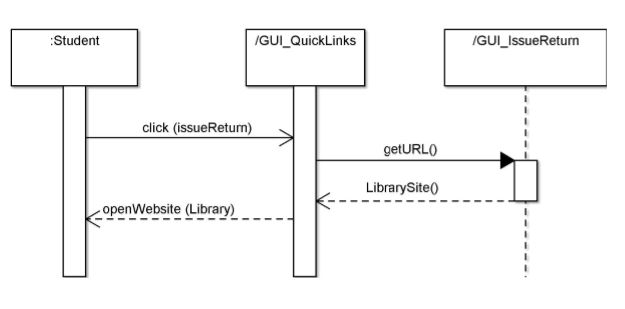
**SIGN UP**

****

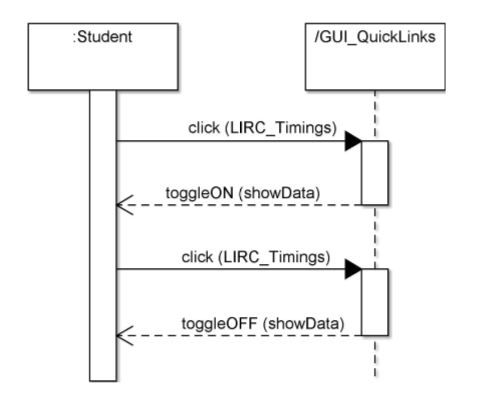
**NAVMENU WITH BOOK REQUISITION FORM AND FEEDBACK FORM**

****

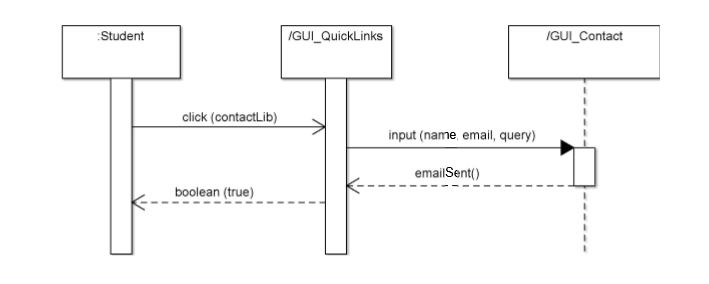
**ISSUE AND RETURN**

****

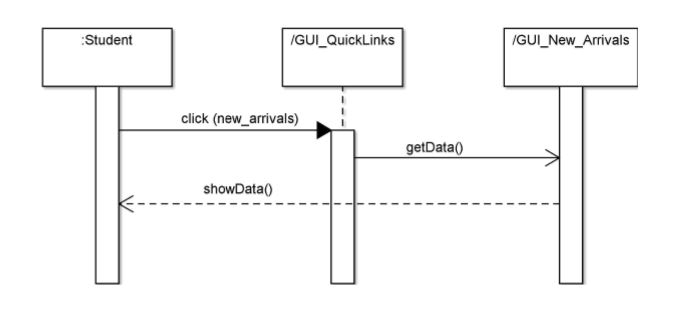
**LIRC TIMINGS**

****

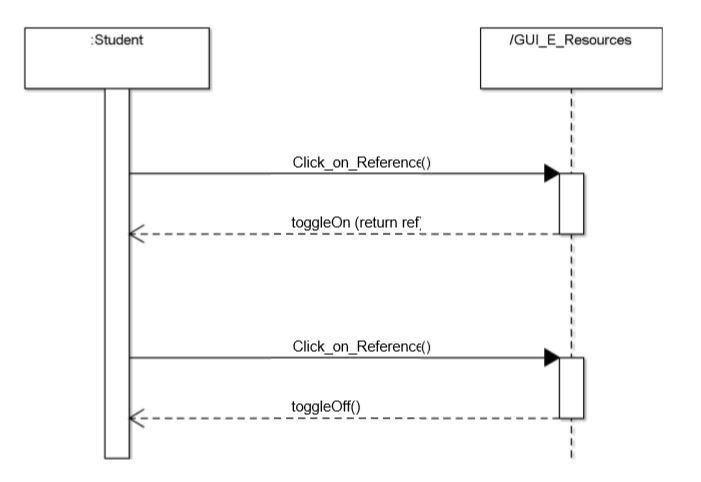
**CONTACT LIBRARIAN**

****

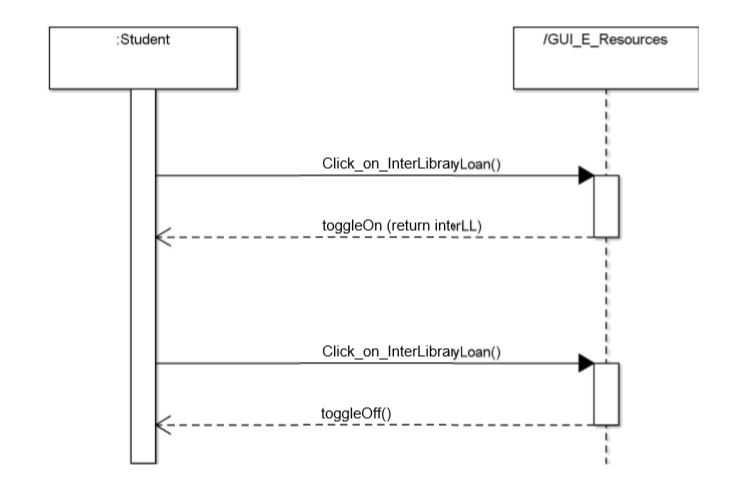
**NEW ARRIVALS**

****

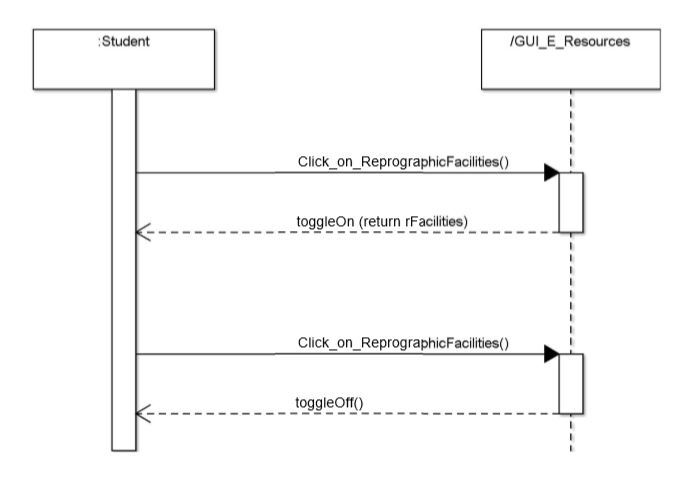
**E-RESOURCES – REFERENCE**

****

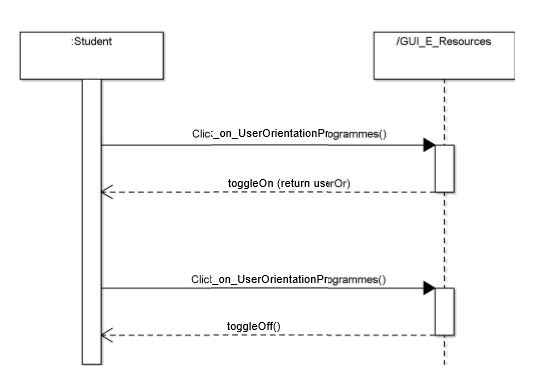
**E-RESOURCES – INTER LIBRARY LOAN**

****

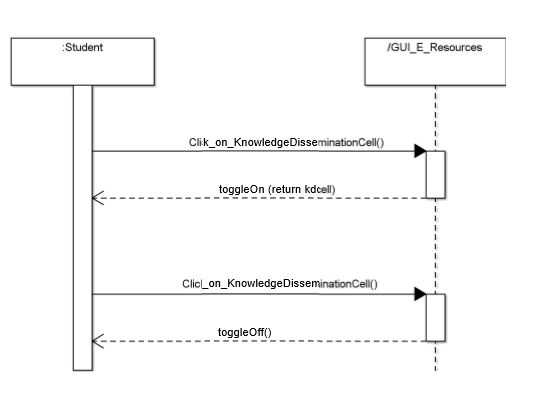
**E-RESOURCES – REPROGRAPHIC FACILITIES**

****

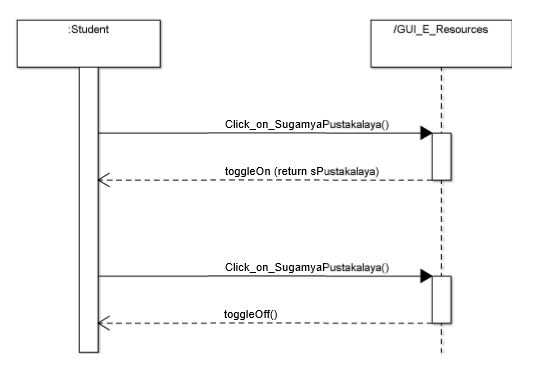
**E-RESOURCES – USER ORIENTATION PROGRAMMES**

****

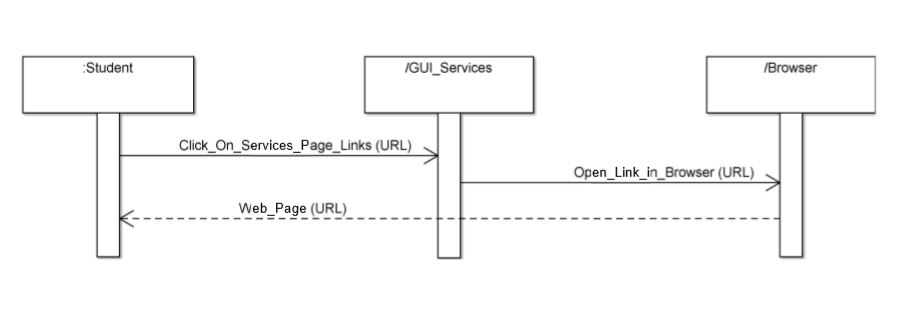
**E-RESOURCES – KNOWLEDGE DISSEMINTAION CELL**

****

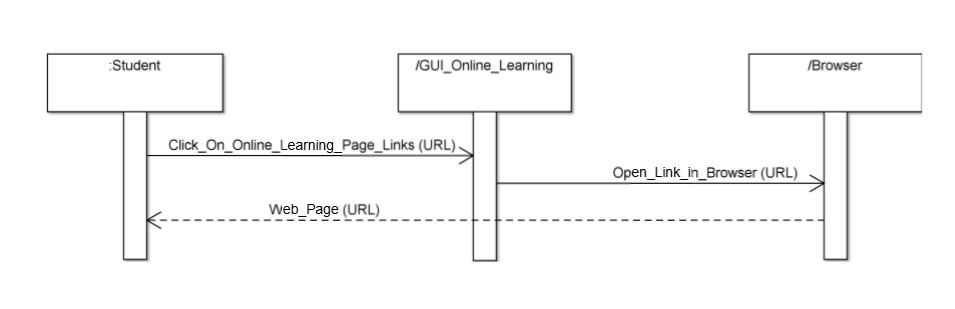
**E-RESOURCES – SUGAMYA PUSTAKALAYA**

****

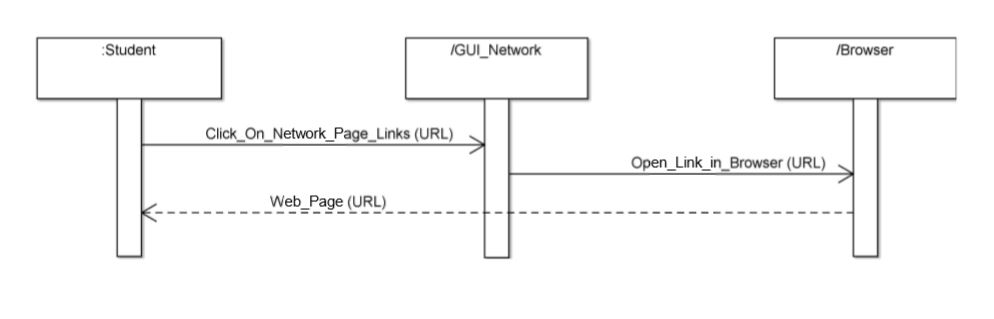
**SERVICES**

****

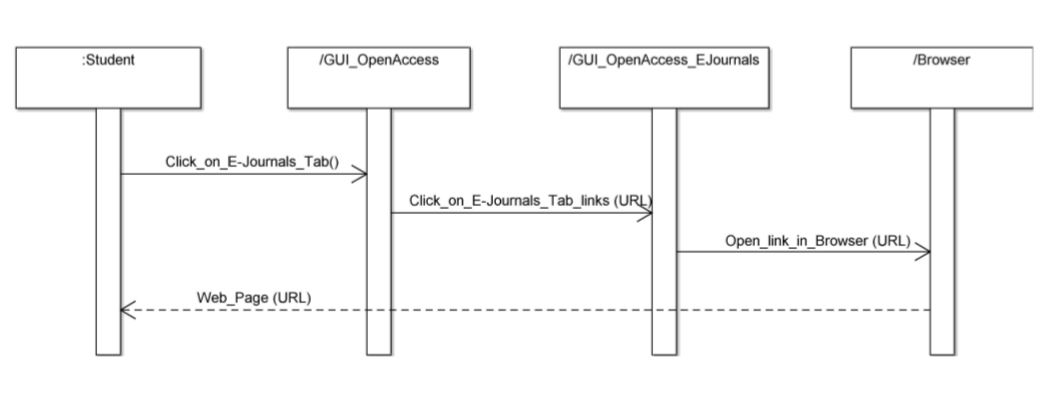
**ONLINE LEARNING**

****

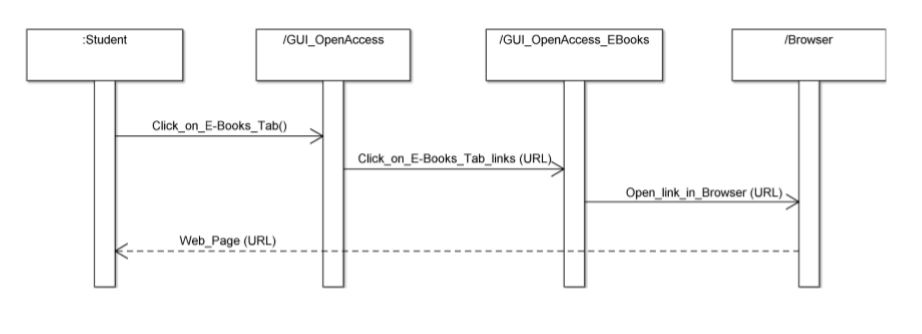
**NETWORK**

****

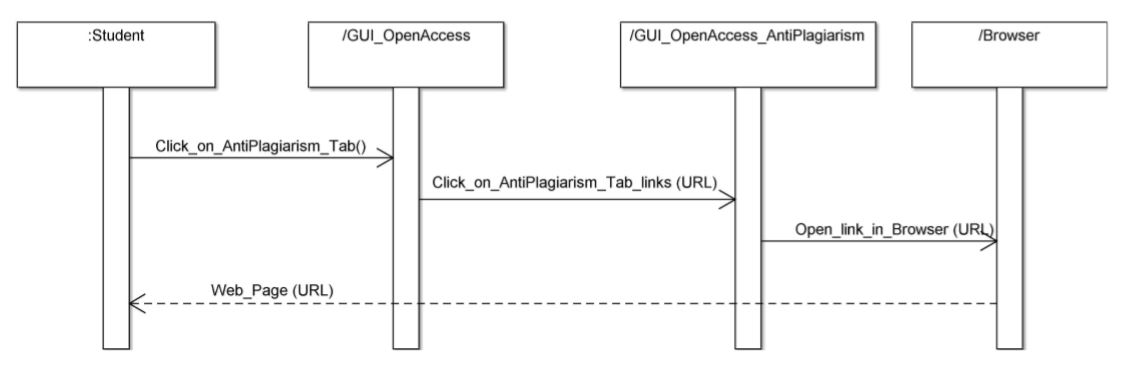
**OPEN ACCESS – E-JOURNALS**

****

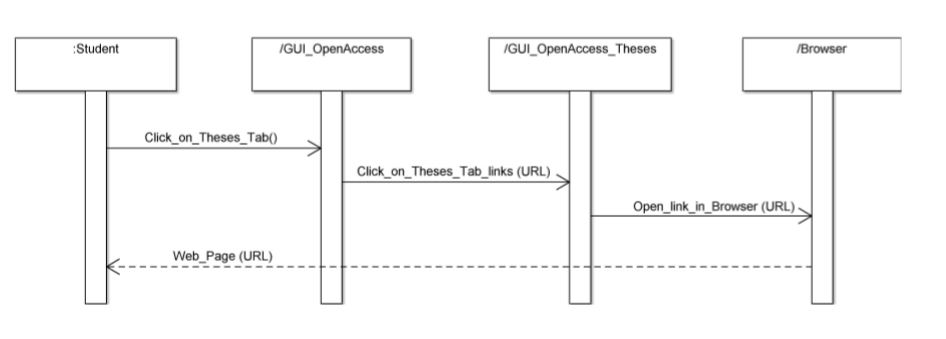
**OPEN ACCESS – E-BOOKS**

****

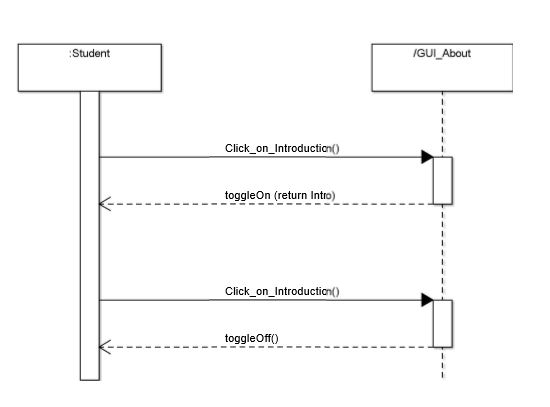
**OPEN ACCESS – ANTI PLAGIARISM**

****

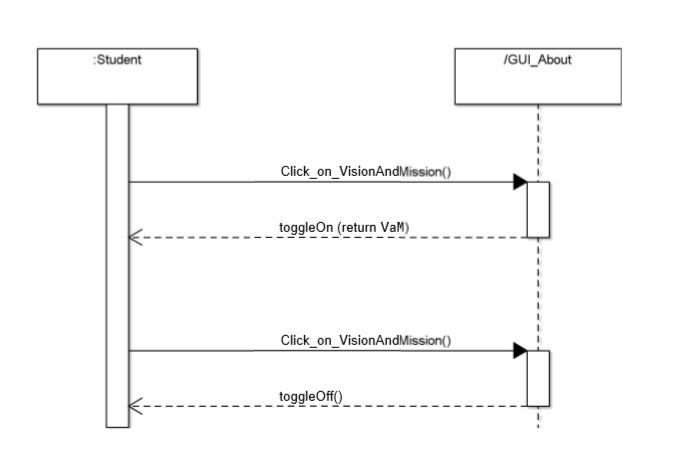
**OPEN ACCESS – THESES**

****

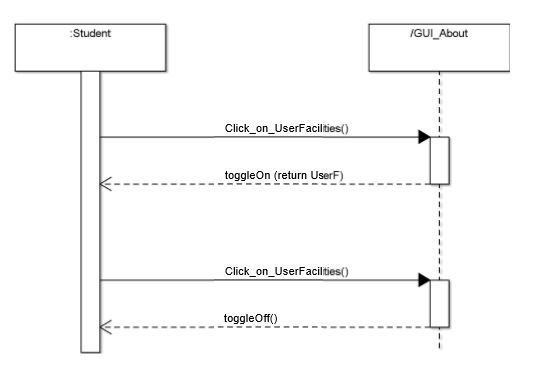
**ABOUT - INTRODUCTION**

****

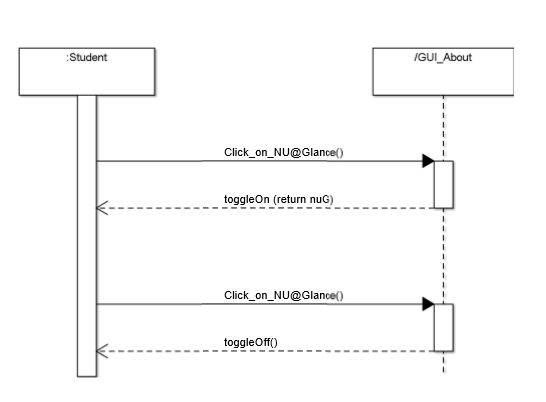
**ABOUT – VISION AND MISSION**

****

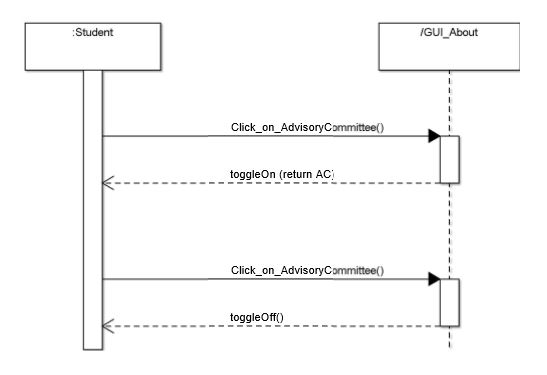
**ABOUT – USER FACILITIES**

****

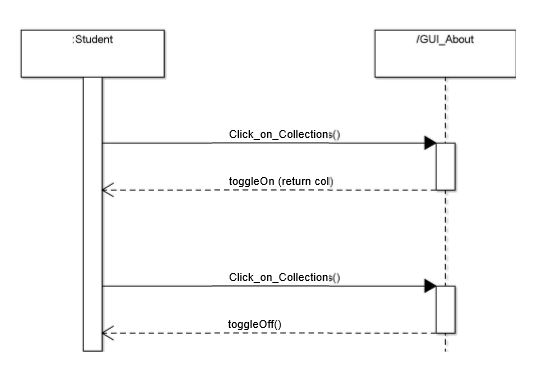
**ABOUT – NU @ A GLANCE**

****

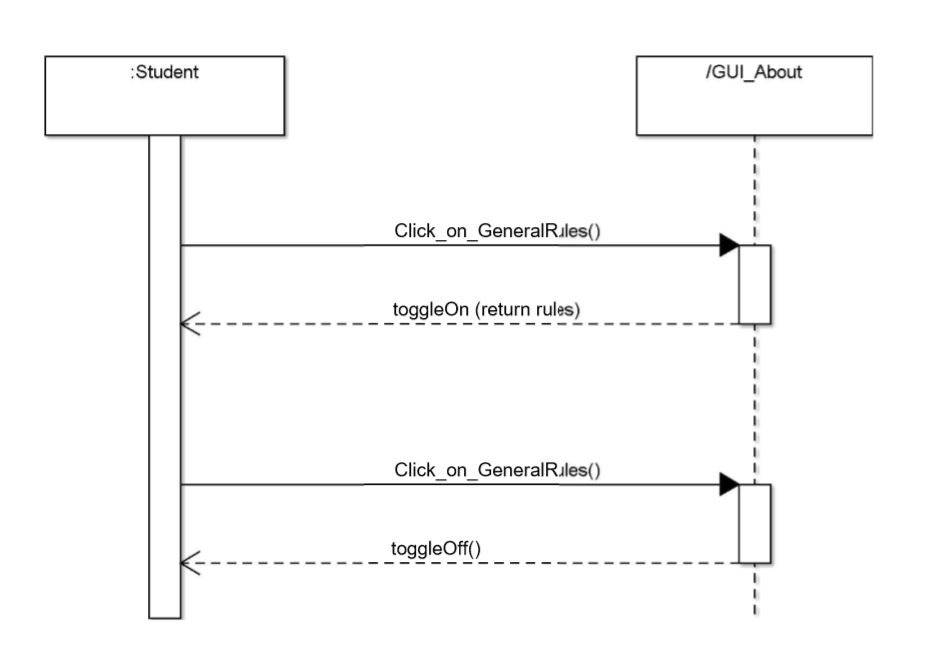
**ABOUT – ADVISORY COMMITTEE**

****

**ABOUT - COLLECTIONS**

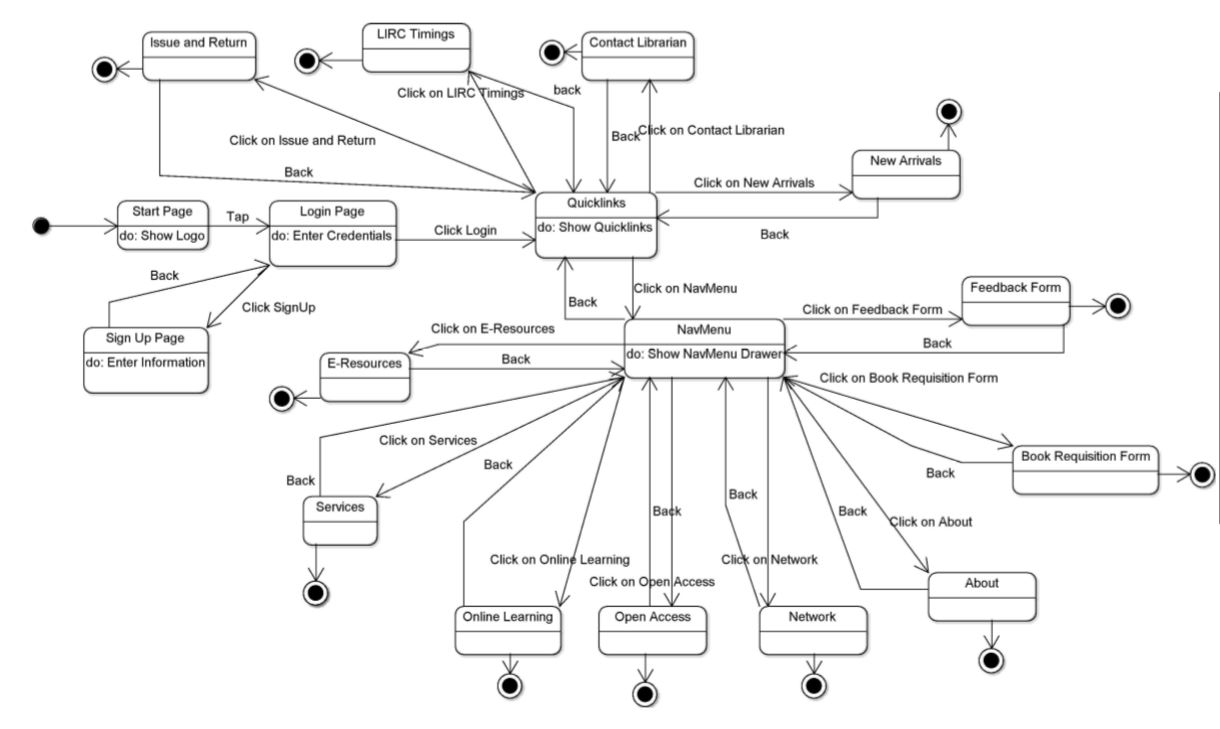
****

**ABOUT – GENERAL RULES**

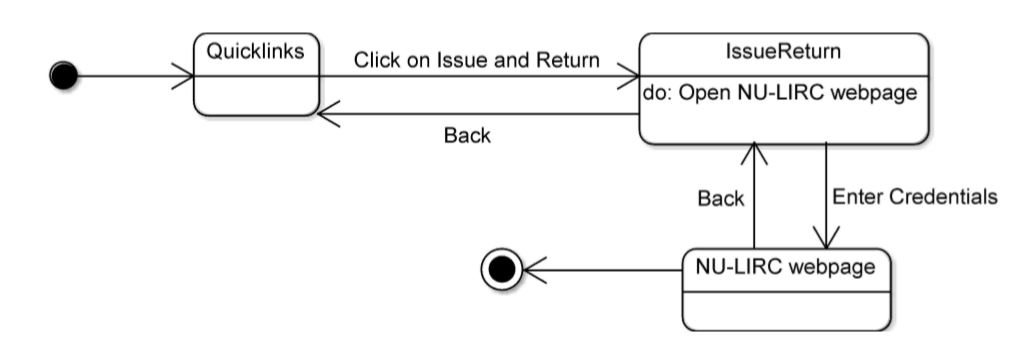
****

**State Diagrams**

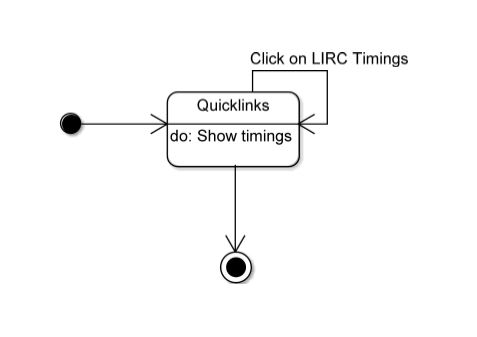
**LOGIN AND SIGN UP**

****

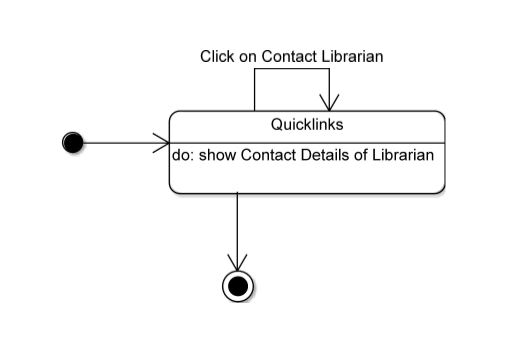
**ISSUE AND RETURN**

****

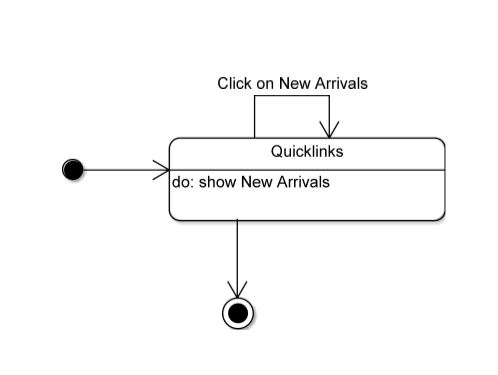
**LIRC TIMINGS**

****

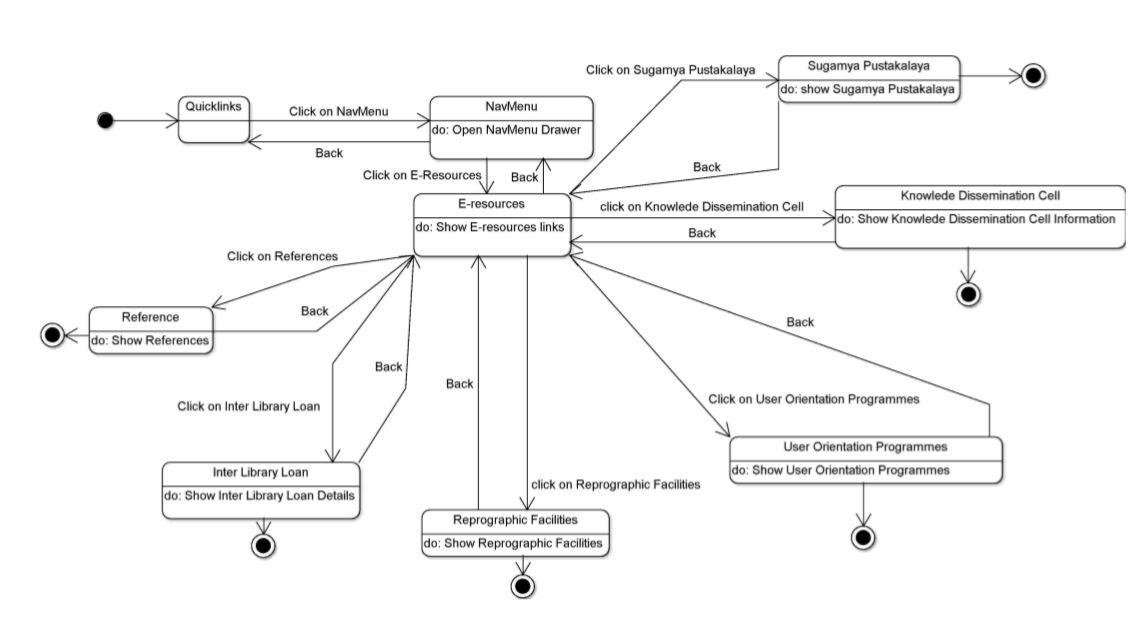
**CONTACT LIBRARIAN**

****

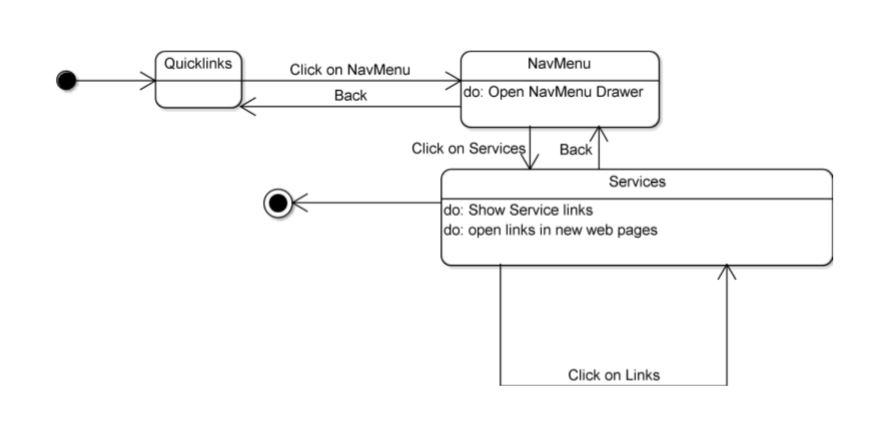
**NEW ARRIVALS**

****

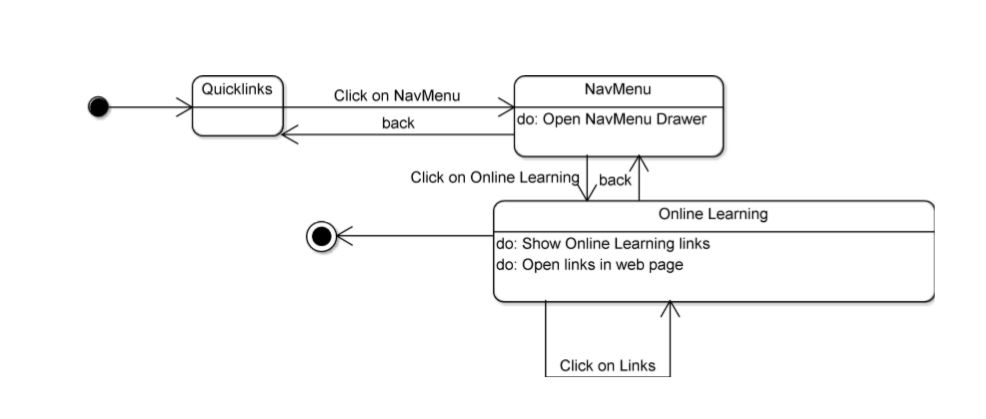
**E-RESOURCES**

****

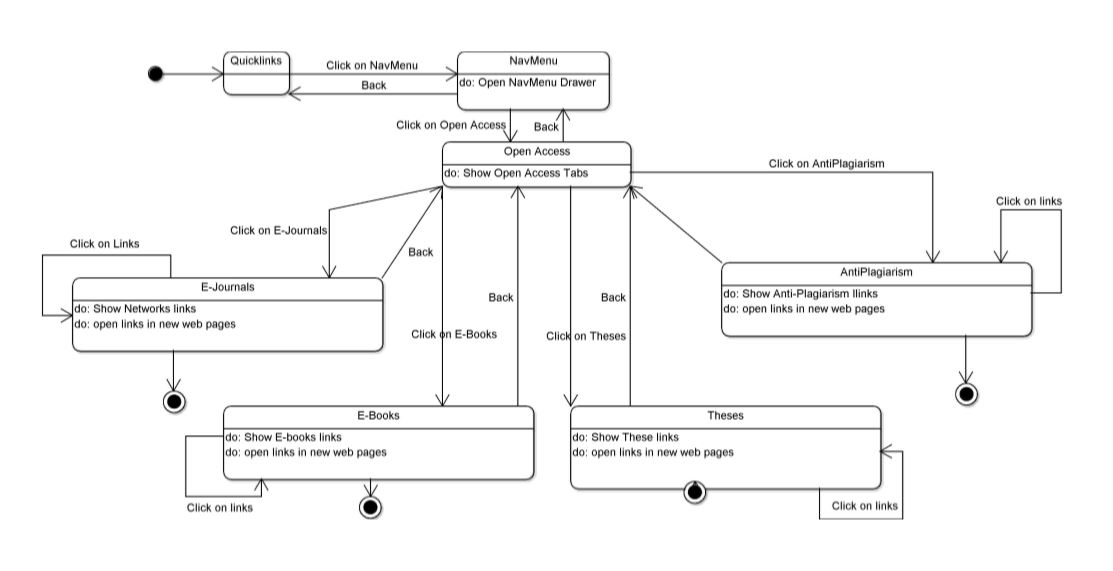
**SERVICES**

****

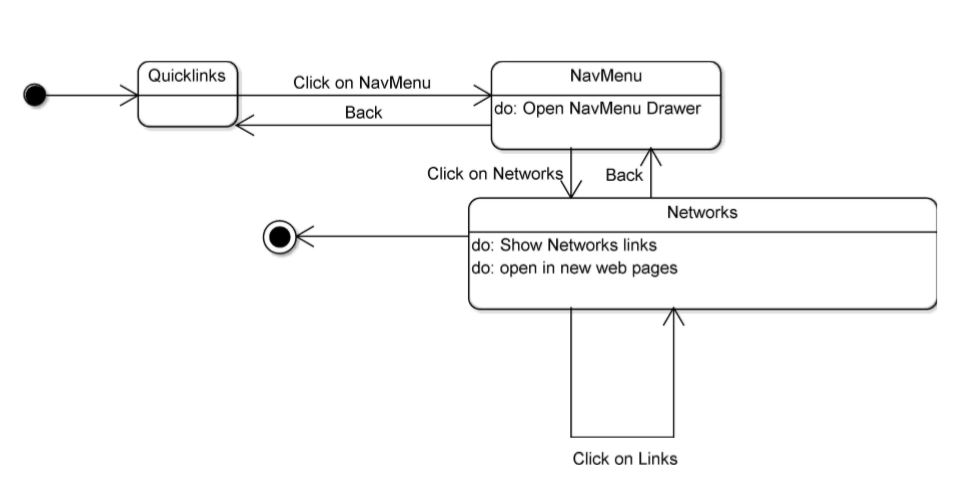
**ONLINE LEARNING**

****

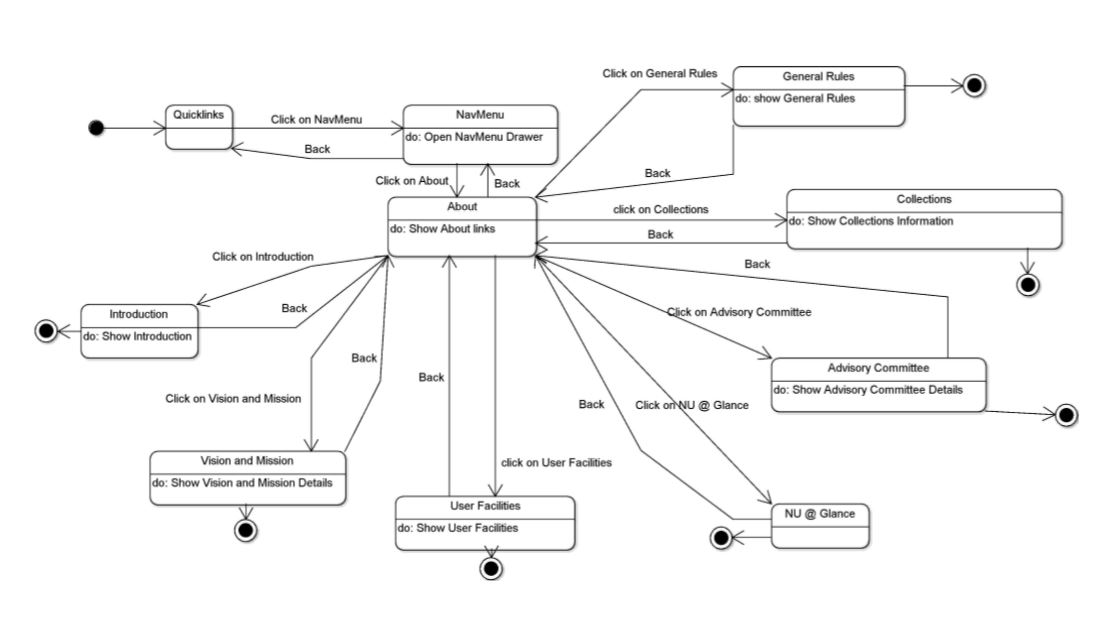
**OPEN ACCESS**

****

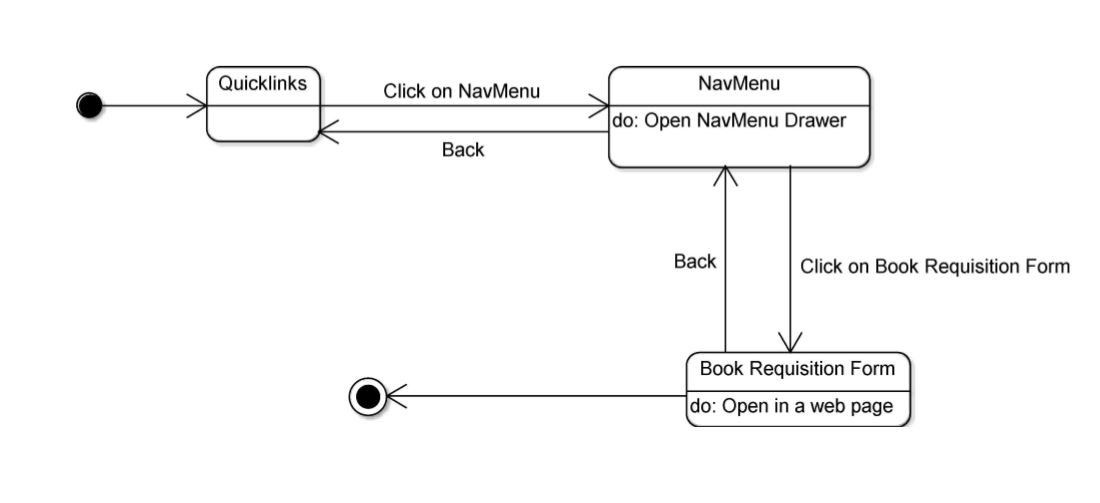
**NETWORK**

****

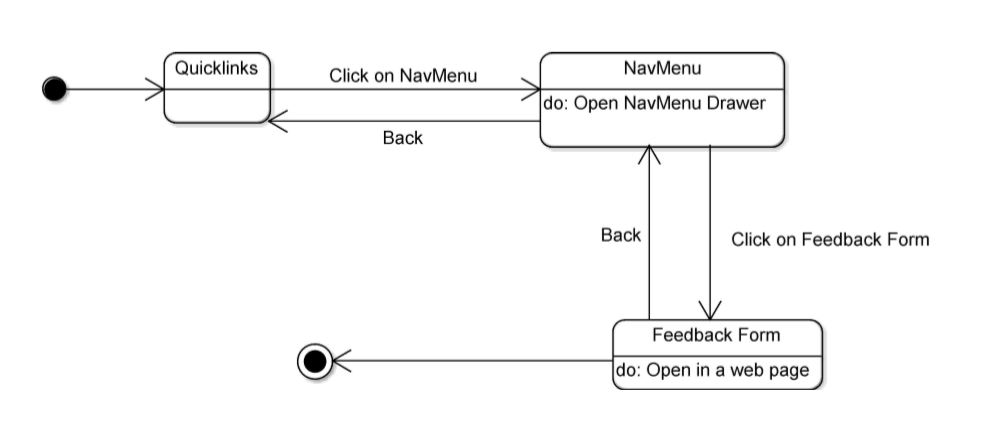
**ABOUT**

****

**BOOK REQUISITION FORM**

****

**FEEDBACK FORM**

****

**3.1 Logical Architecture Description**

**3.1.1 Explanation of class diagram**

Most of the classes extends AppCompatActivity class and implements the Navigation View interface which is being shown by association linkage.

**3.1.2 Explanation of sequence diagram**

Arrow line signifies there is a send message taken place. Response is being shown by dotted arrows.

**3.1.2.1 LOGIN:** It allows students to login with the username and password that are being registered in the database already. It loops being on same page until the correct information is not given.

**3.1.2.2 SIGN UP:** It allows students to sign up for this application.

**3.1.2.3 NAVMENU WITH BOOK REQUISITION FORM AND FEEDBACK FORM:** It allows students to navigate to different tabs – E-Resources, Services, Online Learning, Open Access, Network, About, Book Requisition Form and Feedback.

**3.1.2.4 ISSUE AND RETURN:** It allows students to go the NU-LIRC web site through the web view of the app to check their issued book status and penalties.

**3.1.2.5 LIRC TIMINGS:** It allows students to view the timings of the library.

**3.1.2.6 CONTACT LIBRARIAN:** Shows contact details of the librarian.

**3.1.2.7 NEW ARRIVALS:** Shows new arrivals in the library.

**3.1.2.8-3.1.2.13 E-RESOURCES:** It allows students to view E-resources available through the library.

**3.1.2.14 SERVICES:** It allows students to view Services provided by the library.

**3.1.2.15 ONLINE LEARNING:** It allows students to access online learning links provided by the library.

**3.1.2.16 NETWORK:** It allows students to view the Network of the library.

**3.1.2.17-3.1.2.20 OPEN ACCESS:** It allows students to view E-Journals, E-Books, Anti-Plagiarism and Theses available in the library.

**3.1.2.21-3.1.2.27 ABOUT:** It allows students to information about the library.

**3.2. Class Name : Login**

This class allows the user to enter the system by authenticating the entered credentials.

**3.2.1. Method Name : checkLogin()**

This verifies the credentials with the firebase database.

**3.3. Class Name : SignUpPage**

This class allows the user to sign up for the app by entering the required information.

**3.3.1 Method Name : startRegister()**

This save the information entered to the firebase authentication page.

**3.3.2 Method Name : saveToDatabase()**

This save the information entered to the firebase realtime database page.

**3.4. Class Name : AppCompactActivity**

This is the android studio class that all the other Activities extends (inherits) to be able to display the application on the android system.

**3.4.1. Method Name : onCreate()**

This method is responsible to create the activity when the application is launched on the system.

**3.4.2. Method Name : onStart()**

This is the first method to be executed when the activity opens/reopens.

**3.4.3. Method Name : onBackPressed()**

This method is executed when the back button is pressed o the android system.

**3.4.4. Method Name : onCreateOptionsMenu()**

Output :: This method returns a boolean value

This Method is called upon when the option menu on the top right corner on the app is pressen and any item from the menu is selected.

**3.5. Class Name : MainActivity**

This class is the first activity that is shown every time the application is launched.

**3.6. Class Name : IssueReturn**

This is the Activity that contains the web view to show the library site on the app itself. This Activity is launched whenever we click o the IssueReturn on the Main QuickLinks page.

**3.7. Class Name : EresourcesActivity**

This is the Activity from the Navgation menu which contains the data for the e-resourses of the app.

**3.7.1. Method Name : getData()**

This method extracts the data from the external text files saved in the app itself and displays it on the screen.

**3.8. Class Name : NetworkActivity**

This is the Activity from the Navgation menu which contains the data for the network resources of the app.

**3.9. Class Name : OnlineLearning**

This is the Activity from the Navgation menu which contains the links for the online learning resources of the app.

**3.10. Class Name : AboutActivity**

This is the Activity from the Navgation menu which contains the data of the app and what it contains as well as what are the number of books in the library.

**3.10.1. Method Name : getData()**

This method extracts the data from the external text files saved in the app itself and displays it on the screen.

**3.11. Class Name : ServicesActivity**

This is the Activity from the Navgation menu which contains the data for the services resources of the app.

**3.12. Class Name : OpenAccess**

This is the Activity from the Navgation menu which contains the data for the open access resources of the app.

**3.13. Class Name : QuickLinks**

This is the main page of the application after the login is done. It contains the links to other activities of the app. It also shows the LIRC timings, contact info., new arrivals and a link to IssueReturn.

**3.14. Interface Name : NavigationView**

This is an android implemented interface which we implemment in almost all the class which contains a navgation drawer.

**3.14.1. Method Name : onNavigationItemSelected()**

Output :: This method returns a boolean value

This method contind all the links to the other Activities of the app like the Eresources, Services, etc.

**3.15. Class Name : Application**

Inbuilt Android class.

**3.15.1. Method Name : onCreate()**

This method is responsible to create the activity when the application is launched on the system.

**3.16. Class Name : NULIRCApp**

This class is the base class for the Initialization of Firebase database and authentication.

**3.17. Class Name : Fragment**

This is like a miniature version of an Activity which can be pushed up in an activity as many as you want.

**3.17.1. Method Name : onCreate()**

This method is responsible to create the fragment when the activity is launched on the system.

**3.17.2. Method Name : onCreateView()**

This method is responsible to create the View of the screen when the activity is launched on the system.

**3.17.3. Method Name : onViewCreated()**

This method is responsible for the remaining execution afte the fragment is created.

**3.17. Class Name : FragmentPageAdapter**

This class is extended by TabPageAdapter to be able to display the fragment in a form of a list.

**3.17.1. Method Name : getItem()**

Output :: This returns the fragment selected

This method selects a fragment and displays it to the screen, one at a time.

**3.17.2. Method Name : getCount()**

Output :: This returns the number of fragments

**3.17.3. Method Name : getPageTitle()**

Output :: This returns the title of the fragment

**3.18. Class Name : TabPageAdapter**

This class extend the FragmentPageAdapter and displays the fragments on the screen.

**3.19. Class Name : EBooks**

This is the Fragment which contains the links for the Ebooks resources of the app.

**3.20. Class Name : EJournals**

This is the Fragment which contains the links for the Ejournals resources of the app.

**3.21. Class Name : Theses**

This is the Fragment which contains the links for the Theses resources of the app.

**3.22. Class Name :** **AntiPlagiarism**

This is the Fragment which contains the links for the AntiPlagiarism resources of the app.

**4.0 Execution Architecture**

Runtime environment required is any device supporting Android Operating System with the minimum version of KitKat (API 19), Android Studio as a deployment platform.

**4.1 Reuse and relationships to other products**

This is a new Android application made on the upcoming NU-LIRC website.

**5.0 Design decisions and tradeoffs**

The website could not be integrated properly with the application so the website data is stored on a Firebase database. When a student logs in to the application, all the data from the Firebase database is extracted and loaded into the application. Due to this, the app may take time to load up after the login.

**6.0. Pseudocode for Components:**

**6.0.1. Class Name : MainActivity**

**Method 1 : onCreate()**

1. super.onCreate(savedInstanceState);

2. setContentView(R.layout.activity\_main);

3. libraryApp = stores the id for the textview that reads LibraryApp

4. toContinue = stores the id of textview that reads Click to Continue

5. logo = stores the id of the niit logo image

6. relativeLayout = stores the id of the whole layout

7. animation = creates the animation object

8. load the animation on the libraryApp textview

9. load the animation on the toContinue textview

10. load the animation on the logo imageview

11. set onClickListener on the relativeLayout

1. onClick() : load the next activity (loginPage)

**6.0.2. Class Name : AdminLogin**

**Method 1 : onCreate()**

1. super.onCreate(savedInstanceState);

2. setContentView(R.layout.activity\_main);

3. store the admin username textview id

4. store the admin password textview id

5. get Firebase Instance

6. create new progressDialog object

7. adminButton = store the admin login button id

8. set onClickListener on the adminButton

1. call the checkLogin method

9. store the student login textview

10. set onClickListener on the studentLogin textview

1. onClick() : load the LoginPage activity

**Method 2 : checkLogin()**

1. get the username form the id and convert it to string

2. get the password from the id and convert it to string

3. if username and password are not empty :

1. show the progressDialog

2. signIn with the username and password to Firebase

3. if signIn is successful :

1. start the QuickLinks activity page

2. dismiss dialog

4. else :

1. show error

2. dismiss dialog

5. end if

4. end if

**Method 3 : onBackPressed()**

1. super.onBackPressed()

2. finish this activity

**6.0.3. Class Name : LoginPage**

**Method 1 : onCreate()**

1. super.onCreate(savedInstanceState);

2. setContentView(R.layout.activity\_main);

3. store the student username textview id

4. store the student password textview id

5. get Firebase Instance

6. create new progressDialog object

7. loginButton = store the student login button id

8. set onClickListener on the login button

1. call the check login method

9. store the admin login textview id

10. set onClickListener on the admin login textview

1. onClick() : load the admin Login page

11. store the signup textview id

12. set onClickListener on the signUp textview

1. onClick() : load the signUp activity

**Method 2 : checkLogin()**

1. get the username form the id and convert it to string

2. get the password from the id and convert it to string

3. if username and password are not empty :

1. show the progressDialog

2. signIn with the username and password to Firebase

3. if signIn is successful :

1. start the QuickLinks activity page

2. dismiss dialog

4. else :

1. show error

2. dismiss dialog

5. end if

4. end if

**Method 3 : onBackPressed()**

1. super.onBackPressed()

2. finish this activity

**6.0.4. Class Name : SignUpPage**

**Method 1 : onCreate()**

1. super.onCreate(savedInstanceState);

2. setContentView(R.layout.activity\_main);

3. get Firebase Instance

4. get FirebaseDatabase reference

5. store name textview id

6. store enrollement textview id

7. store username (email) textview id

8. store password textview id

9. create new progressDialog object

10. createButton = store the register button id

11. set onClickListener on the login button

1. call the startRegister method

12. store the have account textview id

13. set onClickListener on the have account textview

1. onClick() : load the Login page activity

**Method 2 : startRegister()**

1. get the username form the id and convert it to string

2. get the password from the id and convert it to string

3. if username and password are not empty :

1. show the progressDialog

2. create account with the username and password to Firebase

3. if signIn is successful :

1. start the LoginPage activity page

2. call saveToDatabase method

3. dismiss dialog

4. else :

1. show error

2. dismiss dialog

5. end if

4. end if

**Method 3 : saveToDatabase()**

1. get name from id

2. get email from id

3. get enrollment from id

4. if none of the fields are empty :

1. get database reference of child users

2. store the data in the database

5. else :

1. show error

6. end if

**6.0.5. Class Name : AboutPage**

**Method 1 : onCreate()**

1. super.onCreate(savedInstanceState);

2. setContentView(R.layout.activity\_main);

3. create toolbar, navigation and toggle object

4. get the ids of the textviews and toggle buttons

5. get Firebase Instance

6. get FirebaseDatabase reference

7. if user not logged in :

1. go back to login page

8. set onClickListener to all the textviews and apply the toggle effect

**Method 2 : getData()**

input : String filename, TextView textview

output : void

1. initialize string text

2. make an inputstream object to open the filename

3. get the data from the inputstream

4. save the data to the textview

**Method 3 : onStart()**

1. super.onStart()

2. mAuth.addAuthStateListener(mAuthListener);

**Method 4 : onBackPressed()**

1. get drawerlayout id

2. if drawer is open :

1. close drawer

3. else :

1. super.onBackPressed()

**Method 5 : onCreateOptionsMenu()**

input : Menu meu

output : boolean value

1. create menu layout

2. return true

**Method 6 : onOptionsItemSelected()**

1. get selected item id

2. if id equals to the settingd id :

1. open the settings activity

3. else if it equals to logout id :

1. mAuth.signOut()

**Method 7 : onNavigationItemSelected()**

1. get selected item id

2. if id equals to eresources id :

1. open the eresources activity

3. if id equals to services id :

1. open the services activity

4. if id equals to onlineLearning id :

1. open the onlineLearning activity

5. if id equals to openAccess id :

1. open the openAccess activity

6. if id equals to network id :

1. open the network activity

7. if id equals to about id :

1. open the about activity

8. if id equals to bookreqform id :

1. open the bookreqform activity

9. if id equals to feedback id :

1. open the feedback activity

**6.0.6. Class Name : Eresources()**

**Method 1 : onCreate()**

1. super.onCreate(savedInstanceState);

2. setContentView(R.layout.activity\_main);

3. create toolbar, navigation and toggle object

4. get the ids of the textviews and toggle buttons

5. get Firebase Instance

6. get FirebaseDatabase reference

7. if user not logged in :

1. go back to login page

8. set onClickListener to all the textviews and apply the toggle effect

**Method 2 : getData()**

input : String filename, TextView textview

output : void

1. initialize string text

2. make an inputstream object to open the filename

3. get the data from the inputstream

4. save the data to the textview

**Method 3 : onStart()**

1. super.onStart()

2. mAuth.addAuthStateListener(mAuthListener);

**Method 4 : onBackPressed()**

1. get drawerlayout id

2. if drawer is open :

1. close drawer

3. else :

1. super.onBackPressed()

**Method 5 : onCreateOptionsMenu()**

input : Menu meu

output : boolean value

1. create menu layout

2. return true

**Method 6 : onOptionsItemSelected()**

1. get selected item id

2. if id equals to the settingd id :

1. open the settings activity

3. else if it equals to logout id :

1. mAuth.signOut()

**Method 7 : onNavigationItemSelected()**

1. get selected item id

2. if id equals to eresources id :

1. open the eresources activity

3. if id equals to services id :

1. open the services activity

4. if id equals to onlineLearning id :

1. open the onlineLearning activity

5. if id equals to openAccess id :

1. open the openAccess activity

6. if id equals to network id :

1. open the network activity

7. if id equals to about id :

1. open the about activity

8. if id equals to bookreqform id :

1. open the bookreqform activity

9. if id equals to feedback id :

1. open the feedback activity

**6.0.7. Class Name : IssueReturn**

**Method 1 : onCreate()**

1. super.onCreate(savedInstanceState);

2. setContentView(R.layout.activity\_main);

3. create toolbar, navigation and toggle object

4. get the ids of the textviews and toggle buttons

5. get Firebase Instance

6. get FirebaseDatabase reference

7. if user not logged in :

1. go back to login page

8. get the webview id

9. set the webview client to a browser

10. load images on webview

11. set javascript enabled on webview

12. load the url

**Method 2 : getData()**

input : String filename, TextView textview

output : void

1. initialize string text

2. make an inputstream object to open the filename

3. get the data from the inputstream

4. save the data to the textview

**Method 3 : onStart()**

1. super.onStart()

2. mAuth.addAuthStateListener(mAuthListener);

**Method 4 : onBackPressed()**

1. get drawerlayout id

2. if drawer is open :

1. close drawer

3. else :

1. super.onBackPressed()

**Method 5 : onCreateOptionsMenu()**

input : Menu meu

output : boolean value

1. create menu layout

2. return true

**Method 6 : onOptionsItemSelected()**

1. get selected item id

2. if id equals to the settingd id :

1. open the settings activity

3. else if it equals to logout id :

1. mAuth.signOut()

**Method 7 : onNavigationItemSelected()**

1. get selected item id

2. if id equals to eresources id :

1. open the eresources activity

3. if id equals to services id :

1. open the services activity

4. if id equals to onlineLearning id :

1. open the onlineLearning activity

5. if id equals to openAccess id :

1. open the openAccess activity

6. if id equals to network id :

1. open the network activity

7. if id equals to about id :

1. open the about activity

8. if id equals to bookreqform id :

1. open the bookreqform activity

9. if id equals to feedback id :

1. open the feedback activity

**6.0.8. Class Name : NetworkActivity()**

**Method 1 : onCreate()**

1. super.onCreate(savedInstanceState);

2. setContentView(R.layout.activity\_main);

3. get Firebase Instance

4. get FirebaseDatabase reference

5. if user not logged in :

1. go back to login page

6. set onClickListener to all the textviews and apply the toggle effect

**Method 2 : onStart()**

1. super.onStart()

2. mAuth.addAuthStateListener(mAuthListener);

**Method 3 : onBackPressed()**

1. get drawerlayout id

2. if drawer is open :

1. close drawer

3. else :

1. super.onBackPressed()

**Method 4 : onCreateOptionsMenu()**

input : Menu meu

output : boolean value

1. create menu layout

2. return true

**Method 5 : onOptionsItemSelected()**

1. get selected item id

2. if id equals to the settingd id :

1. open the settings activity

3. else if it equals to logout id :

1. mAuth.signOut()

**Method 6 : onNavigationItemSelected()**

1. get selected item id

2. if id equals to eresources id :

1. open the eresources activity

3. if id equals to services id :

1. open the services activity

4. if id equals to onlineLearning id :

1. open the onlineLearning activity

5. if id equals to openAccess id :

1. open the openAccess activity

6. if id equals to network id :

1. open the network activity

7. if id equals to about id :

1. open the about activity

8. if id equals to bookreqform id :

1. open the bookreqform activity

9. if id equals to feedback id :

1. open the feedback activity

**6.0.9. Class Name : OpenAccess()**

**Method 1 : onCreate()**

1. super.onCreate(savedInstanceState);

2. setContentView(R.layout.activity\_main);

3. create toolbar, navigation and tab object

4. create a viewpager and tabAdapter object

5. get the ids of the textviews and toggle buttons

6. get Firebase Instance

7. get FirebaseDatabase reference

8. if user not logged in :

1. go back to login page

9. set onClickListener to all the textviews and apply the toggle effect

**Method 2 : onStart()**

1. super.onStart()

2. mAuth.addAuthStateListener(mAuthListener);

**Method 3 : onBackPressed()**

1. get drawerlayout id

2. if drawer is open :

1. close drawer

3. else :

1. super.onBackPressed()

**Method 4 : onCreateOptionsMenu()**

input : Menu meu

output : boolean value

1. create menu layout

2. return true

**Method 5 : onOptionsItemSelected()**

1. get selected item id

2. if id equals to the settingd id :

1. open the settings activity

3. else if it equals to logout id :

1. mAuth.signOut()

**Method 6 : onNavigationItemSelected()**

1. get selected item id

2. if id equals to eresources id :

1. open the eresources activity

3. if id equals to services id :

1. open the services activity

4. if id equals to onlineLearning id :

1. open the onlineLearning activity

5. if id equals to openAccess id :

1. open the openAccess activity

6. if id equals to network id :

1. open the network activity

7. if id equals to about id :

1. open the about activity

8. if id equals to bookreqform id :

1. open the bookreqform activity

9. if id equals to feedback id :

1. open the feedback activity

**6.0.10. Class Name : OnlineLearning()**

**Method 1 : onCreate()**

1. super.onCreate(savedInstanceState);

2. setContentView(R.layout.activity\_main);

3. create toolbar, navigation and toggle object

4. get the ids of the textviews and toggle buttons

5. get Firebase Instance

6. get FirebaseDatabase reference

7. if user not logged in :

1. go back to login page

8. set onClickListener to all the textviews to open the text in the browser

**Method 2 : onStart()**

1. super.onStart()

2. mAuth.addAuthStateListener(mAuthListener);

**Method 3 : onBackPressed()**

1. get drawerlayout id

2. if drawer is open :

1. close drawer

3. else :

1. super.onBackPressed()

**Method 4 : onCreateOptionsMenu()**

input : Menu meu

output : boolean value

1. create menu layout

2. return true

**Method 5 : onOptionsItemSelected()**

1. get selected item id

2. if id equals to the settingd id :

1. open the settings activity

3. else if it equals to logout id :

1. mAuth.signOut()

**Method 6 : onNavigationItemSelected()**

1. get selected item id

2. if id equals to eresources id :

1. open the eresources activity

3. if id equals to services id :

1. open the services activity

4. if id equals to onlineLearning id :

1. open the onlineLearning activity

5. if id equals to openAccess id :

1. open the openAccess activity

6. if id equals to network id :

1. open the network activity

7. if id equals to about id :

1. open the about activity

8. if id equals to bookreqform id :

1. open the bookreqform activity

9. if id equals to feedback id :

1. open the feedback activity

**6.0.11. Class Name : ServicesActivity()**

**Method 1 : onCreate()**

1. super.onCreate(savedInstanceState);

2. setContentView(R.layout.activity\_main);

3. create toolbar, navigation objects

4. get Firebase Instance

5. get FirebaseDatabase reference

6. if user not logged in :

1. go back to login page

7. set onClickListener to all the textviews and apply the toggle effect

**Method 2 : onStart()**

1. super.onStart()

2. mAuth.addAuthStateListener(mAuthListener);

**Method 3 : onBackPressed()**

1. get drawerlayout id

2. if drawer is open :

1. close drawer

3. else :

1. super.onBackPressed()

**Method 4 : onCreateOptionsMenu()**

input : Menu meu

output : boolean value

1. create menu layout

2. return true

**Method 5 : onOptionsItemSelected()**

1. get selected item id

2. if id equals to the settingd id :

1. open the settings activity

3. else if it equals to logout id :

1. mAuth.signOut()

**Method 6 : onNavigationItemSelected()**

1. get selected item id

2. if id equals to eresources id :

1. open the eresources activity

3. if id equals to services id :

1. open the services activity

4. if id equals to onlineLearning id :

1. open the onlineLearning activity

5. if id equals to openAccess id :

1. open the openAccess activity

6. if id equals to network id :

1. open the network activity

7. if id equals to about id :

1. open the about activity

8. if id equals to bookreqform id :

1. open the bookreqform activity

9. if id equals to feedback id :

1. open the feedback activity

**6.0.12. Class Name : QuickLinks**

**Method 1 : onCreate()**

1. super.onCreate(savedInstanceState);

2. setContentView(R.layout.activity\_main);

3. call the setLIRCTimings

4. create toolbar, navigation and toggle object

5. get Firebase Instance

6. get FirebaseDatabase reference

7. if user not logged in :

1. go back to login page

8. get issueReturn textview id

9. set onClickListener on the issueReturn textview :

1. onCLick() : open the issueReturn activity

**Method 2 : setLIRCTimings()**

1. initialize string text

2. get the all textview id for the displayed data

3. set the data stored in the strings.xml file on the textviews

**Method 3 : onStart()**

1. super.onStart()

2. mAuth.addAuthStateListener(mAuthListener);

**Method 4 : onBackPressed()**

1. get drawerlayout id

2. if drawer is open :

1. close drawer

3. else :

1. super.onBackPressed()

**Method 5 : onCreateOptionsMenu()**

input : Menu meu

output : boolean value

1. create menu layout

2. return true

**Method 6 : onOptionsItemSelected()**

1. get selected item id

2. if id equals to the settingd id :

1. open the settings activity

3. else if it equals to logout id :

1. mAuth.signOut()

**Method 7 : onNavigationItemSelected()**

1. get selected item id

2. if id equals to eresources id :

1. open the eresources activity

3. if id equals to services id :

1. open the services activity

4. if id equals to onlineLearning id :

1. open the onlineLearning activity

5. if id equals to openAccess id :

1. open the openAccess activity

6. if id equals to network id :

1. open the network activity

7. if id equals to about id :

1. open the about activity

8. if id equals to bookreqform id :

1. open the bookreqform activity

9. if id equals to feedback id :

1. open the feedback activity

**7.0 Appendices (if any)**

**NA**