PUZZLE

Mert ÇALIK - 1679406

Oğuz Can KARTAL – 1746106

Berke Çağkan TOPTAŞ – 1746387

Burak VELİOĞLU - 1679257

SOFTWARE DESIGN DESCRIPTION

VERSION 1.1

İçindekiler

1. INTRODUCTION

1.1. Purpose

The purpose of this document is to help the reader visualize the solution to the MOBCOLL project. This document verifies how the design meets the requirements specified in the SRS document through design viewpoints. The design viewpoints will cover all design elements presented before. By using information from IEEE Std 10161998 (scaled down version for CENG 490) this document will provide a direct approach to the development of this project hence reducing feature creep and determine the quality of the design.

1.2. Scope of Project

This document gives a detailed description of MOBCOLL project and the solutions are described by using UML diagrams. This documentation also shows the user interface, the class diagrams, which give the basic structure of the project. This document also gives a clear understanding of how the project will be implemented. The contents of this document do not cover a fully functional system and can include some assumptions which will become certain after the implementation stage.

1.3. Overview

The next chapter, the System Overview section, of this document gives an overview of the functionality of the project. It describes the informal design generally, which technologies will be used etc.

The third chapter, System Architecture section, of this document explains the relationships between the modules to achieve the complete functionality of the system, provides a decomposition of the subsystems in the architectural design and discusses the rationale for selecting the architecture described, including critical issues and trade/offs that were considered.

The fourth chapter, Data Design section, of this document explains how the information domain of the system is transformed into data structures and lists the objects and its attributes, methods and method parameters.

The fifth chapter, Component Design section, of this document takes a closer look at what each component does in a more systematic way.

The sixth chapter, Human Interface Design section, of this document describes the functionality of the system from the user's perspective and displays screenshots showing the interface from the user's perspective.

The seventh and last chapter, Requirements Matrix section, of this document provides a cross-reference that traces components and data structures to the requirements in our SRS document.

1.4. Reference Material

http://www.ceng.metu.edu.tr/_media/course/ceng490/sdd_template.pdf?rev=

2. SYSTEM OVERVIEW

2.1. Description of Problem

People may be collecting audio albums because of their collecting hobby or their interest in music. Audio albums in physical form are often provided with decorative covers (cover art). Since there are numerous audio albums and a new album come onto the market nearly every week keeping record of these albums is not very easy. Therefore, it is critical to keep record of all the albums collectors have.

2.2. Technologies Used

This system will be programmed by using Java through Eclipse tool which is a software development environment comprising an integrated development environment (IDE). We use MYSQL for the database management and communication with database. The HTTP protocol will be used to facilitate communications between the client and server. Jason objects are the communication format between the client and server.

2.3. Application Overview

MOBCOLL Software will be a mobile application that is designed for collectors which try to keep record of their collection items and interact with other collectors. The goal of this project is to provide an easy way to keeping record of own albums and also finding out the destination and owner of album being looked for.

3. SYSTEM ARCHITECTURE

3.1. Architectural Design

Users interact with Android User Interface when the application starts; Android User Interface invokes the server side by sending requests. Server side gets the request and creates appropriate queries and sends these queries to database. From database correct results are returned. Server side returns these results to the client side in proper format and Android User Interface shows these results to the user.

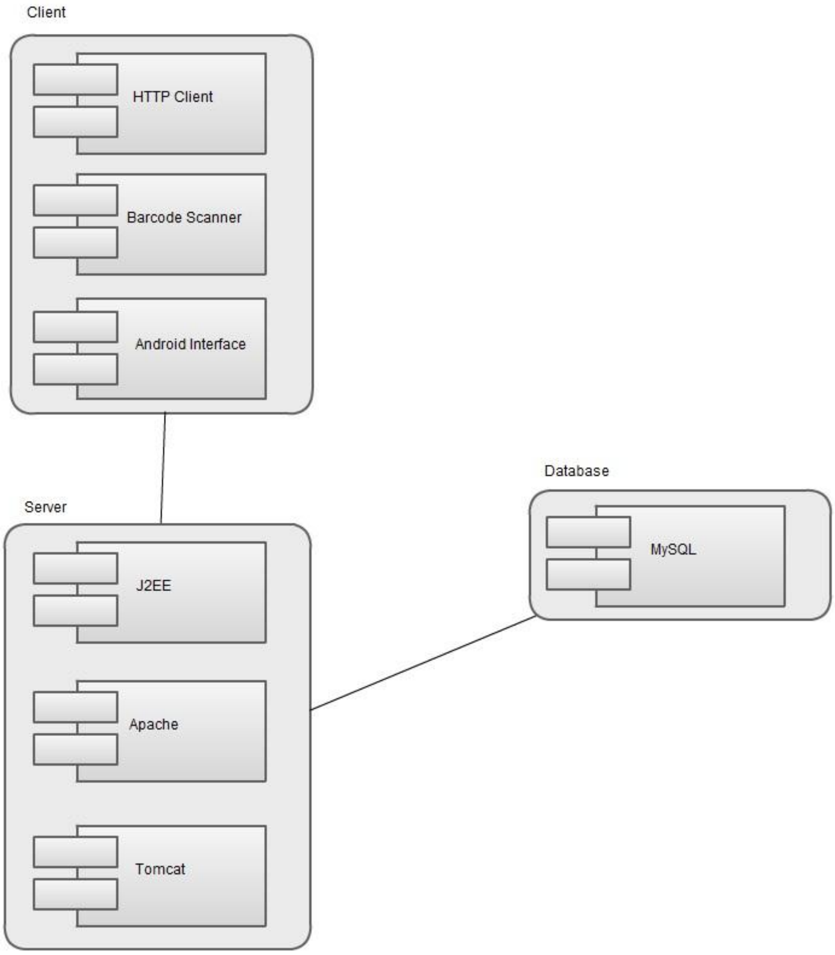


Figure 1 - Deployment Diagram of MOBCOLL

3.2. Decomposition Description

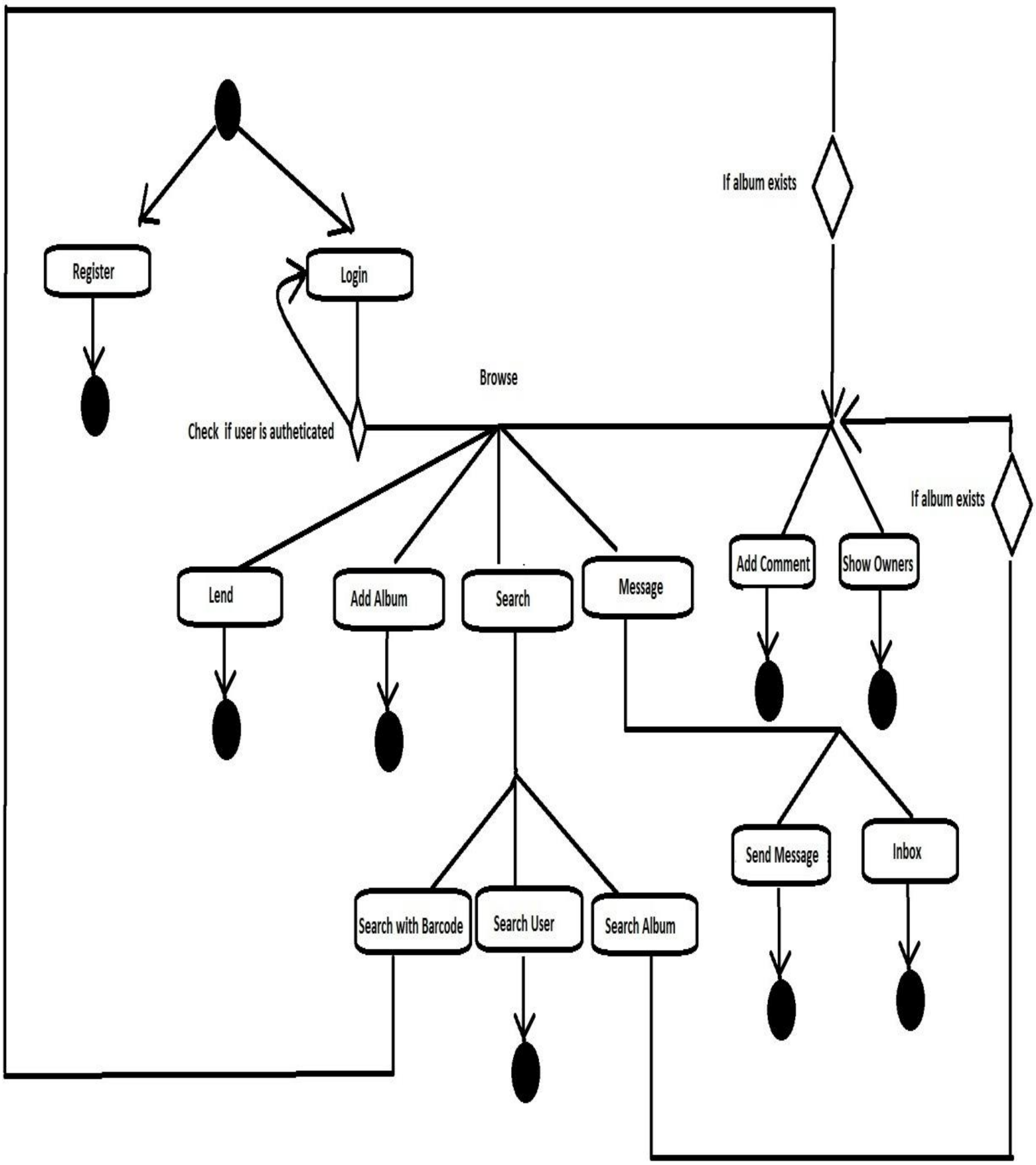
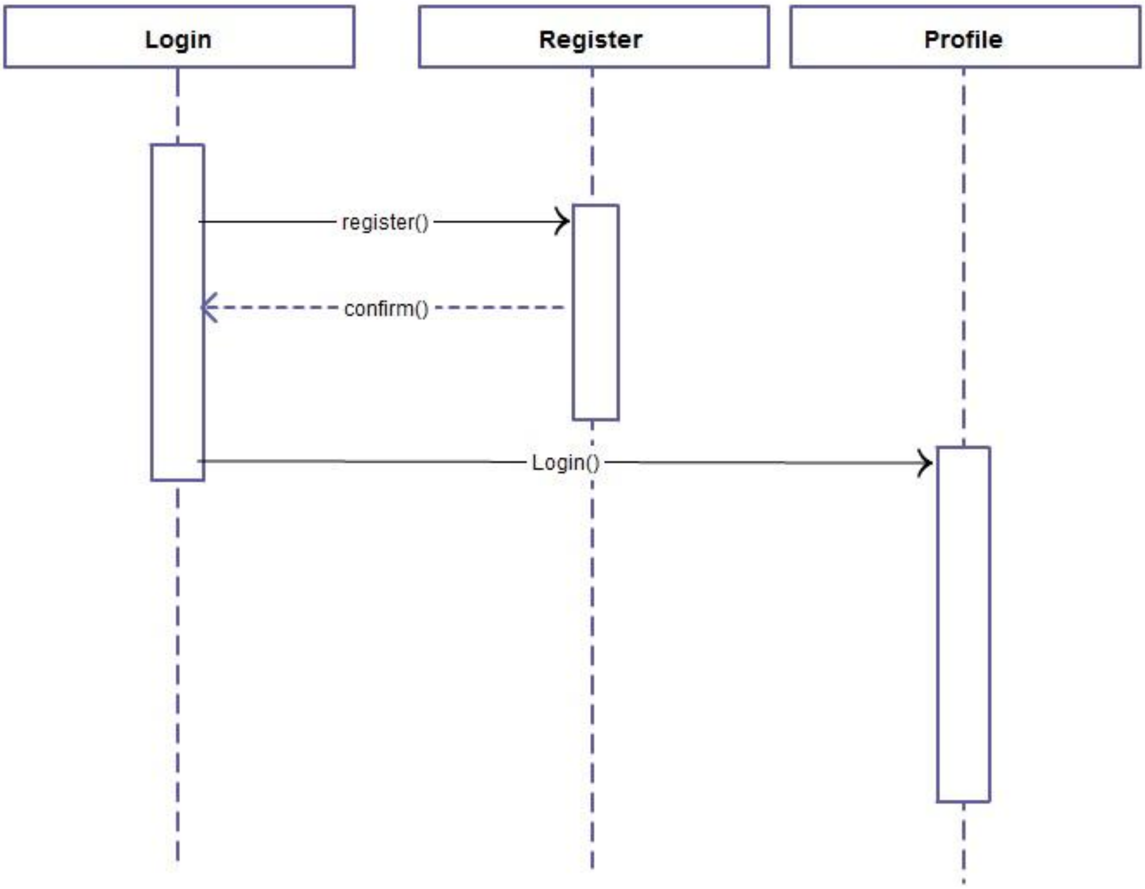


Figure 2 - Data Flow Diagram of MOBCOLL

Application opens with “login page” on the screen. If user has an already existing account, he/she can login to the system by entering username and password correctly and finally clicking to the “login” button. Another option is to login to system with already existing Facebook account. User can click “log in with Facebook” button and he/she is directed to login page of Facebook acoount. When the user is logged in to Facebook he/she is automatically logged in to our system and directed to profile page. If user does not have an account, he/she is directed to the “register page” where user can enter to by clicking “register” button. In “register page” user enters username, password, name, surname, gender, mail address and birthdate and when this information is valid a new account is created for user and redirected to the “login page”.



s

Figure 3 - Login Sequence Diagram

After the user logged in to the system he/she is directed to own profile page. In this page user information and owned albums are displayed. Albums can be seen in more than one categories. User can add a new album to his/her collection, set his/her albums as lent,create new categories for albums, move an album to those album categories, send a message to another user and search an album or a user in "profile page".

To add a new album "add album" button should be clicked. User can send a request to server by entering album name or scanning barcode with mobile phone's camera. If album already exists in database it is added instantly, if not admins process the request and if the given information is valif album is added to both user's albums' and album database.

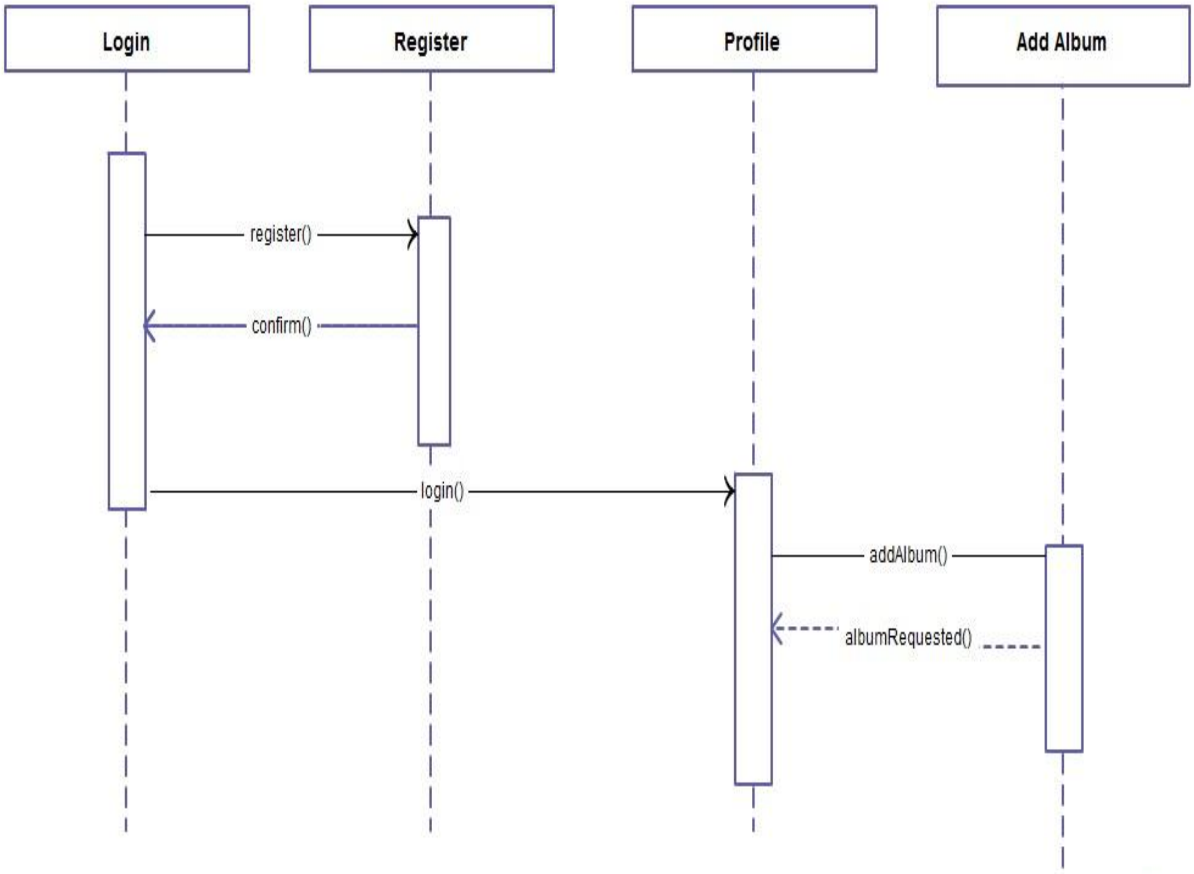


Figure 4 - Add Album Sequence Diagram

To set an album as lent user should extend an album by clicking “extend” button. After clicking that button to album is extended and its’ information is displayed to the user. After that user can set the album’s lending information by entering borrower’s user name, lending date and due date of getting back the album.

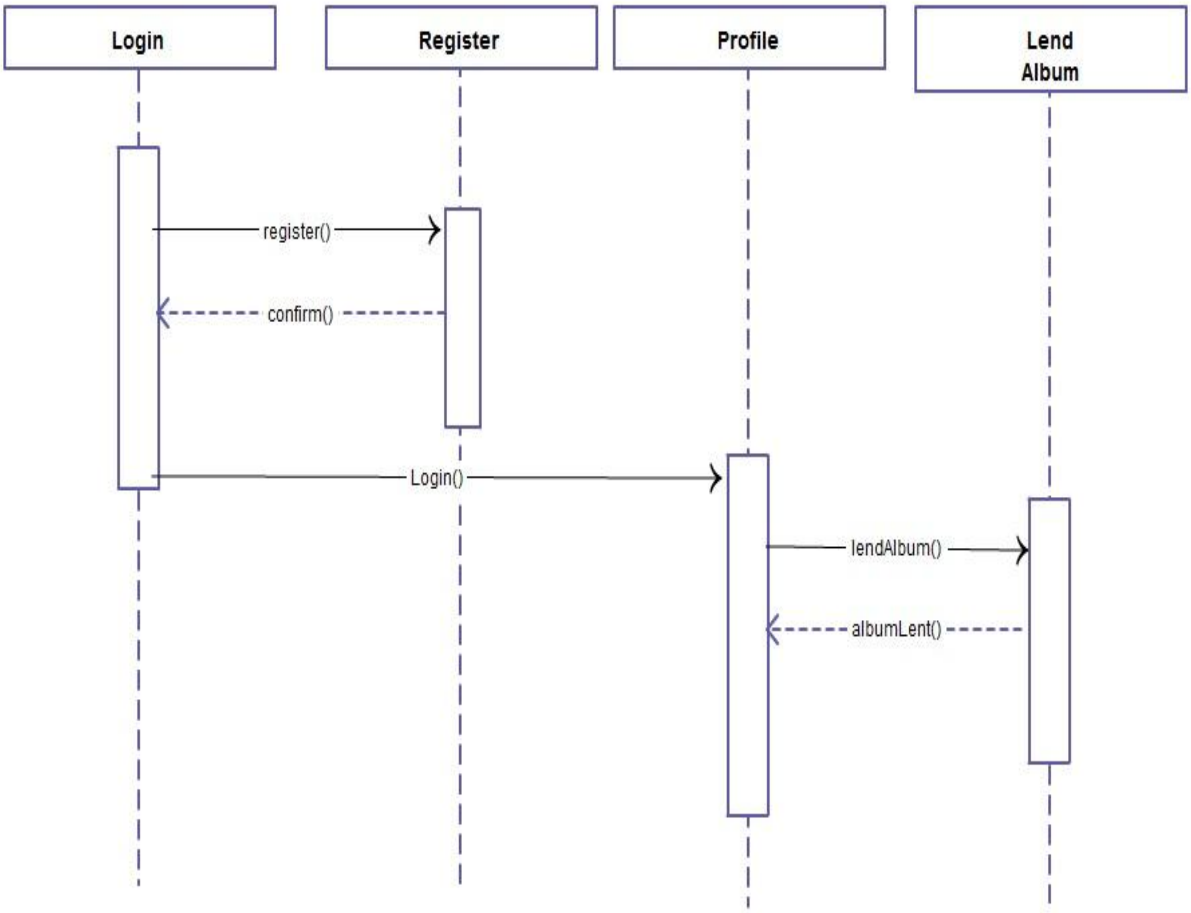


Figure 5 - Lending Sequence Diagram

To create a new category "create a new category" button from the menu should be clicked. After that a textbox pops up and the desired name of the category is written by user so a new category has been created.

When an album is added by user it exists in "all albums section" from here there are two different ways to add albums to categories. If user holds down his finger on the desired album the list of all existing categories is shown, from there user can select the categories he/she wants to move selected album. The other option is; if a user is in one of the created categories he/she can click the "insert " button and the list of all owned albums is shown, from there user can select the albums he/she wants to move to selected category.

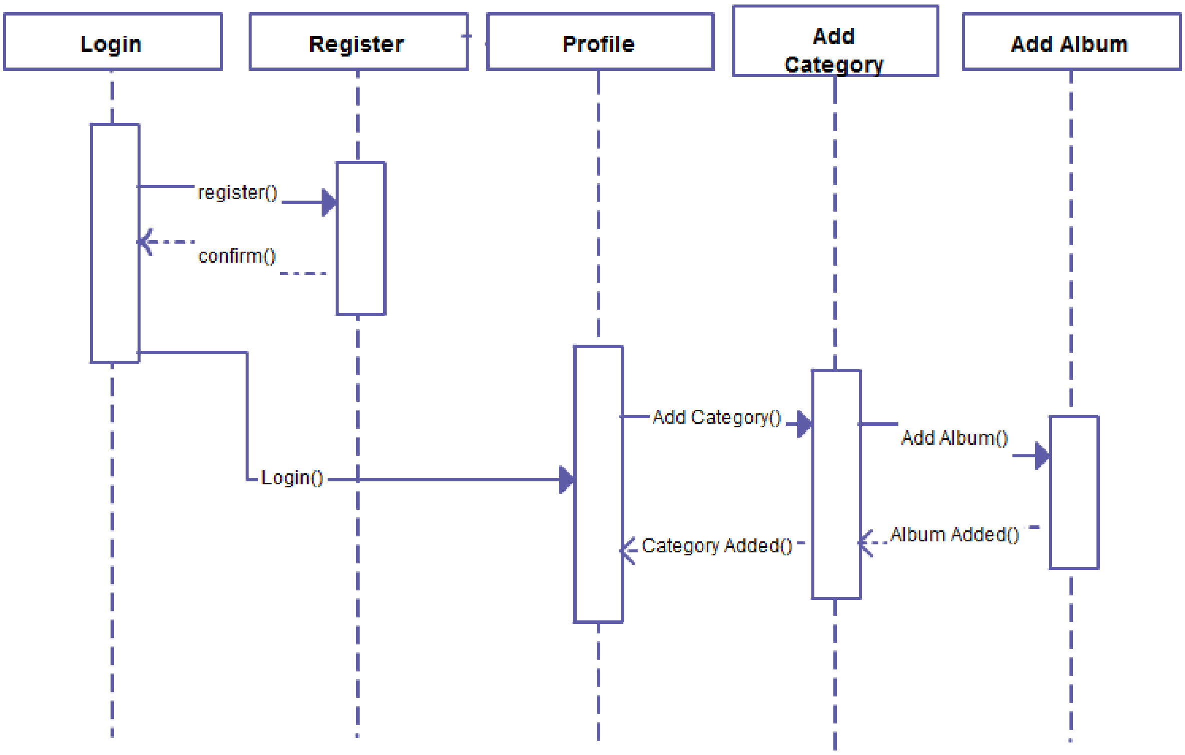


Figure 6 - Categorize Sequence Diagram

To send message to another user there are 2 different ways. First, user can go to "message page" from "profile page" directly by clicking "messages" button. The other choice is going to "message page" from "album page". In "album page" user can click on any owners' username to go to "message page". After entering the "message page" user writes the message and reciever's username and click the "send" button.

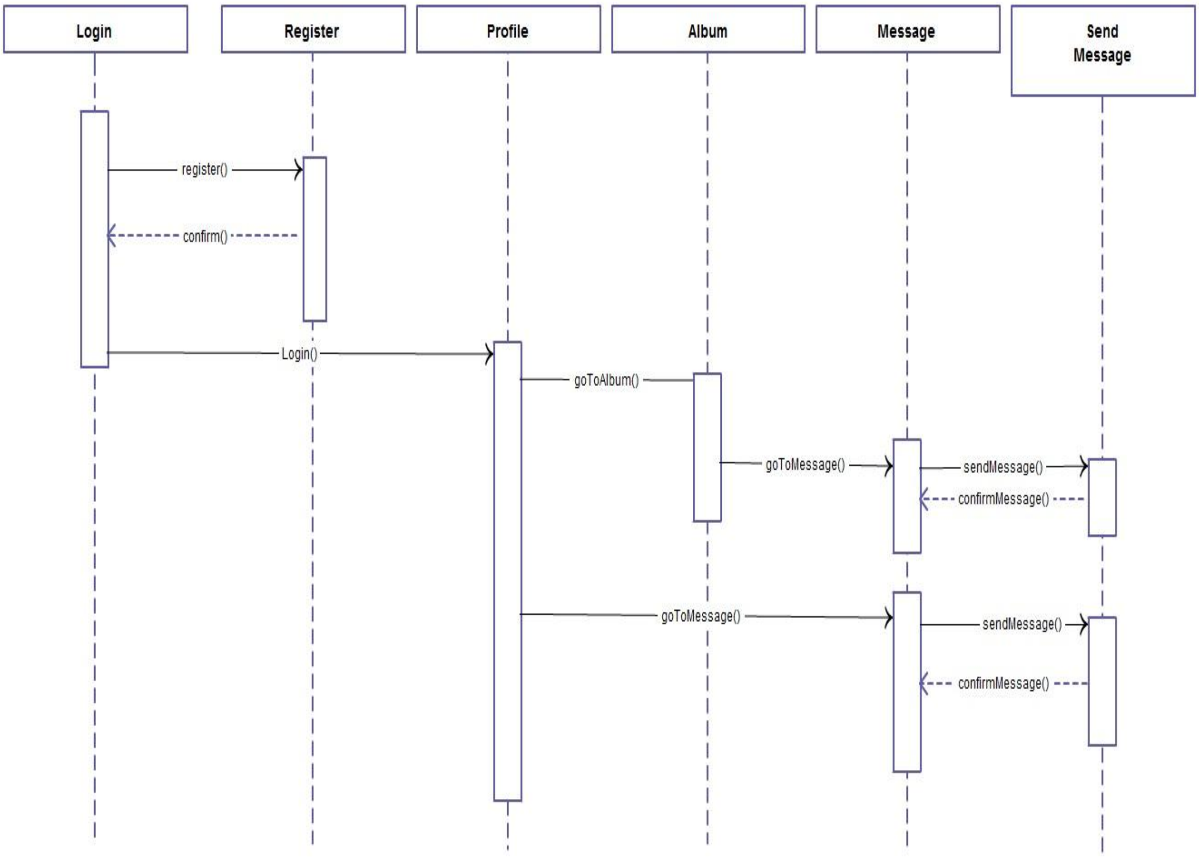


Figure 7 - Send Message Sequence Diagram

To search an album or user there are 2 different ways. User can go to “search page” by clicking the “search” button which exists in both “profile page” and “album page”. After entering the “search page” user can search an album by it’s barcode or other album information like album name, one of its song’s name, artist name, genre or release year. User can also search another user by username, name or surname and take a glance at his/her profile page.

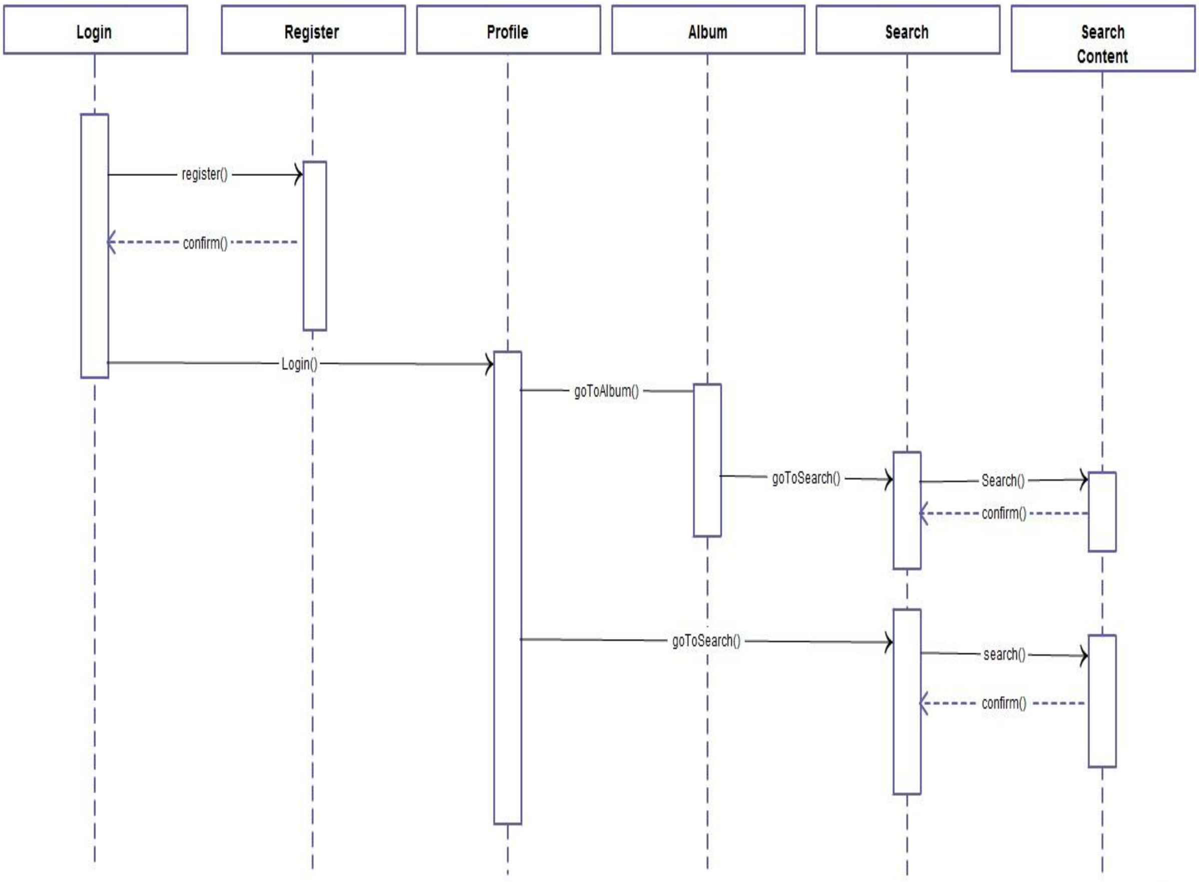


Figure 8 - Search Sequence Diagram

4. Data Diagram

4.1. Data Description

In this Project, MySQL Database is used to store the data. There are five tables in database which are User, Album, Song, Artist and Message tables. ER Diagram of the database is given below, detailed information which may not be understood easily from the diagram also elaborated.

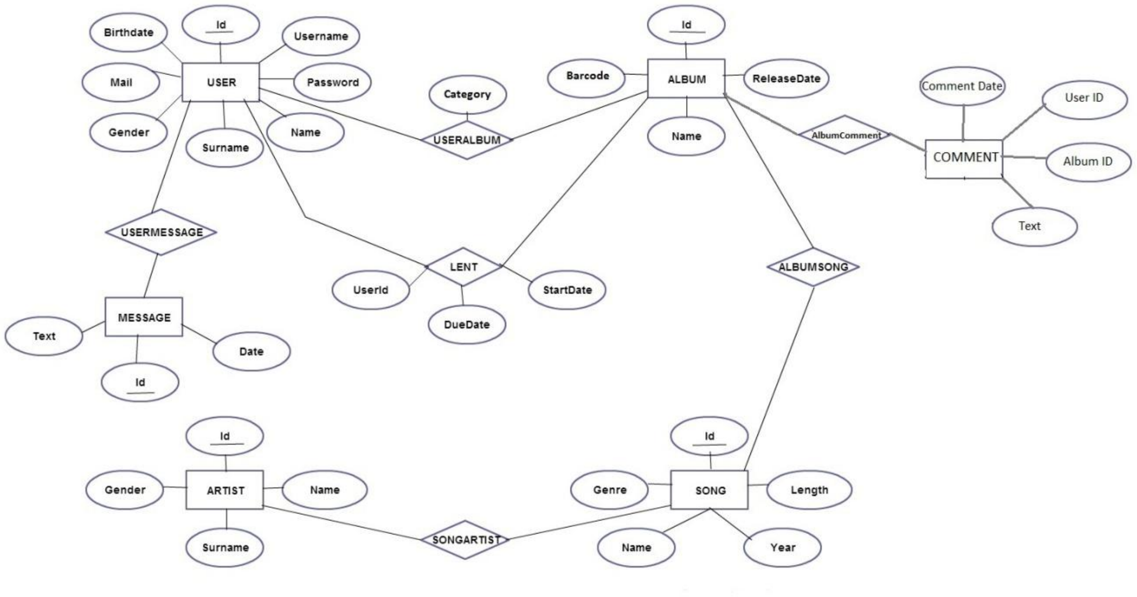


Figure 9 - ER Diagram of MOBCOLL

In User table information of users is hold which belongs to registered users. Each User has different Username and database assigns different Ids for all users.

In Album table Release Date, Name and Barcode information are stored. Different Ids are given for all albums. Content of the album is not an attribute of this table because same song can be member of different albums. Therefore, there is a Song table to hold them.

In Song table different songs from all album tables are stored with their length, year, name and genre information. Artist information is not stored in that table; because there may be more than one artist sing that song. Therefore, there is an Artist table to hold singers.

In Artist table name, surname and gender information are stored for all artists. Different Ids are given to each artist which is used to link artist table with song table.

In Message table, text and date information of messages are stored which are sent between the registered users. Id attribute is also hold to distinguish messages from each other.

4.2. Data Dictionary

There are two main parts in the project which are namely server and client sides.

4.2.1. Server Side

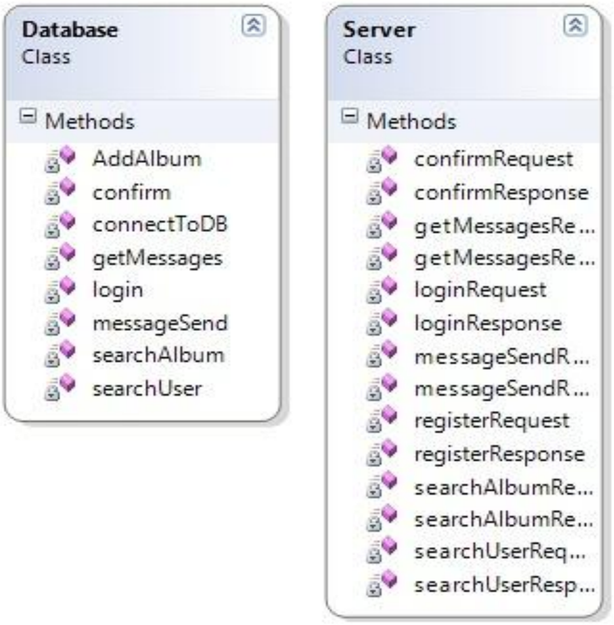


Figure 10 - Server Class Diagram

In server side, Database class is used to manage database operations of the system. All requests which are made by clients related with database handled with this class by the server side. Database class is used by Server class, which is the main class of the server, to do those operations. Server class gets HttpRequests from the clients, process these requests and response related documents as HttpResponse. Class diagram of the server side is given above.

4.2.2. Client Side

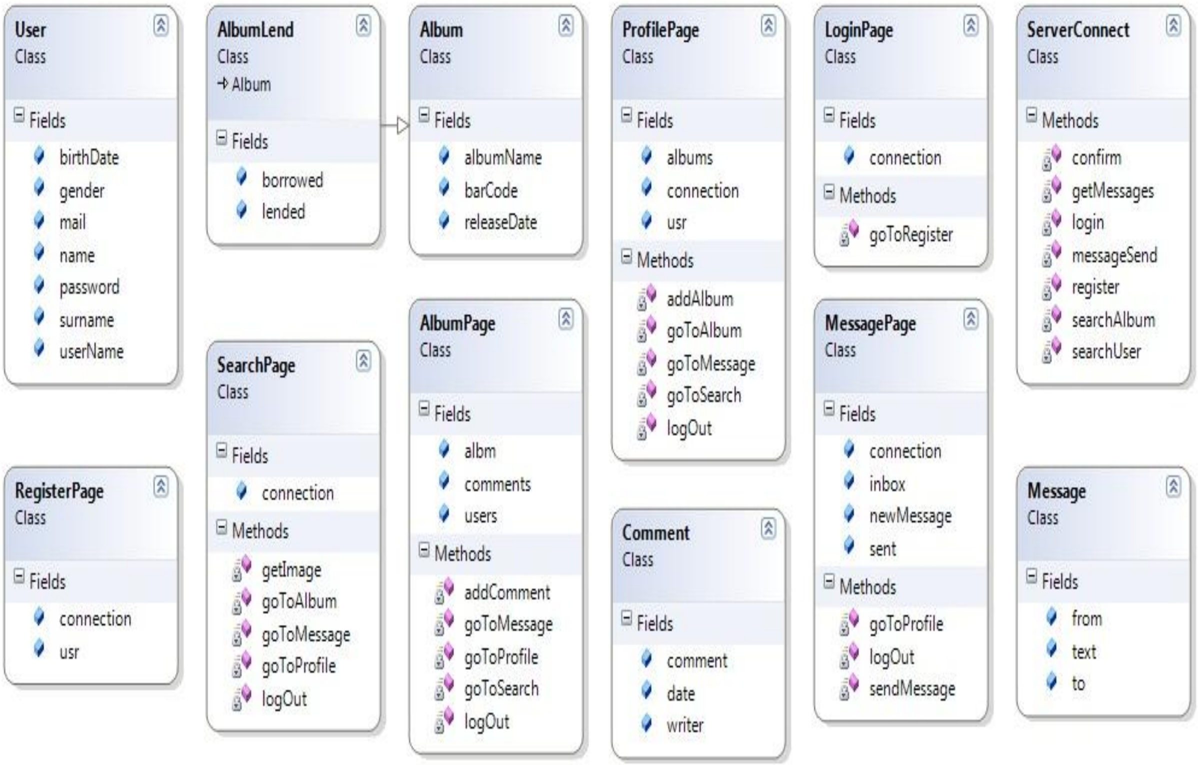


Figure 11 - Client Class Diagram

In Client Side of the project, there are six class which stand for the pages of the user-interface. These are namely LoginPage, RegisterPage, SearchPage, AlbumPage, ProfilePage and MessagePage classes. User, Album, AlbumLend, Comment and Message classes are hold data structures which are used by page classes. ServerConnect class provides the connection between server and client sides. Class diagram of the client side is given above.

5. Component Design

There are three main components in the project to maintain the process of the system. Deployment Diagram of components are given in Figure 1.

5.1. Server Component

In that component two classes are used.

5.1.1. Database Class

- Confirm() : This method is used to confirm the register request of the client by controlling the information given by user with user table.
- AddAlbum() : This method is used to report new album request to admin of the system.
- ConnectToDB() : This method connects to MySQL database with given username and password.
- GetMessages() : This method returns messages which are sent or received by related user.
- Login() : This method is used to control the information which are sent in the period of the login section.
- MessageSend() : This method compounds the message and the user in the database.
- SearchAlbum() : This method searches the album table in the database according to the parameters given and returns the results.
- SearchUser() : This method searches the user table in the database according to the parameters given and returns the results.

5.1.2. Server Class

- AddAlbumRequest() : This method gets the album request from the client and use the AddAlbum() method of the Database class.
- ConfirmRequest() : This method gets the confirm request for the registration from the client and use the confirm() method of the Database class.
- ConfirmResponse() : This method sends the confirm response to the client.
- GetMessageRequest() : This method gets the get message request from the client and use the getMessages() method of the Database to obtain all messages.
- GetMessageResponse() : This method sends the messages to the client.
- LoginRequest() : This method gets the login information from the client and use the login() method of the Database class.
- LoginResponse() : This method send accept or reject answer to the client.
- MessageSendRequest() : This method use messageSend method from database class and compounds the message and the user in the database.
- MessageSendResponse(): This method returns bool whether message is sent or not.
- searchAlbumRequest() : This method use searchAlbum method from database class and searches the album table in the database according to the parameters given and returns the results.
- searchAlbumResponse() : This method response the output of searchAlbumRequest() to the client side.
- searchUserRequest() : This method use searchUser method from database class searches the user table in the database according to the parameters given and returns the results.
- searchUserResponse(): This method response the output of searchUserRequest method to the client side.

5.2. Client Component

5.2.1. SearchPage Class

- GetImage(): This method access the camera of the phone and execute the barcode reader.
- GoToAlbum(): This method is executed when user clicks to the album name. It opens the album page of selected album name.
- GoToMessage(): This method is executed when the user clicks message button which is next to the name of the user and opens the compose message screen.
- GoToProfile(): This method opens the profile page on screen.
- Logout(): This method opens the login page.

5.2.2. Album Page

- AddComment(): This method sends server a comment structure which is written in textbox in album page.
- GoToMessage(): This method is executed when the user is selected from the owner list of that album.
- GoToProfile(): This method opens the profile page on screen.
- GoToSearch(): This method opens search page on screen.
- Share(): This method share the album through Facebook.
- Logout():This method opens the login page.
- AddAlbum() : This method sends the album add request to the server.

5.2.3. Profile Page

- AddAlbum() : This method sends an album add request to the server with album barcode parameter.
- GoToAlbum():This method is executed when user clicks to the album name. It opens the album page of selected album name.
- AddCategory(): This method creates a new category.
- InsertAlbum(): This method selected album to selected category.
- GoToMessage():This method is executed when the user is selected from the owner list of that album.
- GoToSearch():This method opens search page on screen.
- Logout():This method opens the login page.

5.2.4. Login Page

- goToRegister(): This method opens the register page to allow user register to the system.
- goToProfile(): This method used to login to the system and opens the profile page.

5.2.5. Message Page

- goToProfile():This method used to login to the system and opens the profile page.
- LogOut():This method opens the login page.
- SendMessage(): This method sends private message to the selected user.

5.2.6. Server Connect

- Confirm(): This method sends the registration information to the server and get confirmation.
- GetMessages(): This method gets the private message sent to user from server.
- GetAlbum():This method gets the selected album from server.
- Login(): This method control the information which are given in the login page through the server.
- MessageSend(): This method sends the message text to the server which is delivered to the target user by server.
- AddAlbum(): This method sends album add request to the server.
- searchAlbum(): This method sends search parameters for the album search to the server.
- searchUser(): This method sends search parameters for the user search to the server.

6. HUMAN INTERFACE DESIGN

6.1. Overview of User Interface

6.1.1. Login Page

In login page there are two text boxes and three buttons. Text boxes are for entering username and password. First button is login button and the second one is register button. The last and the third button is logging in to server through Facebook.

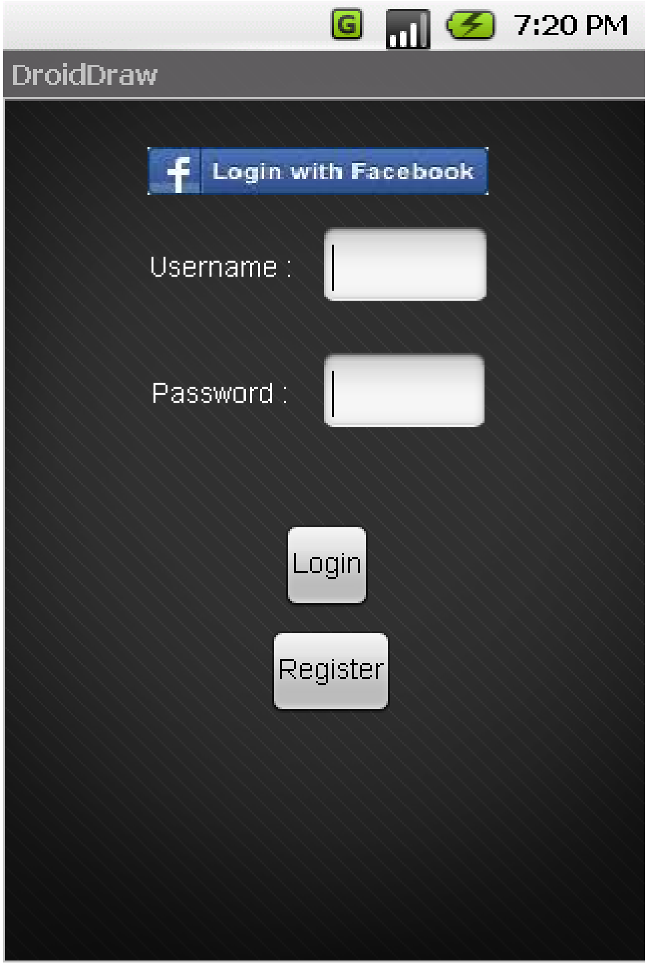
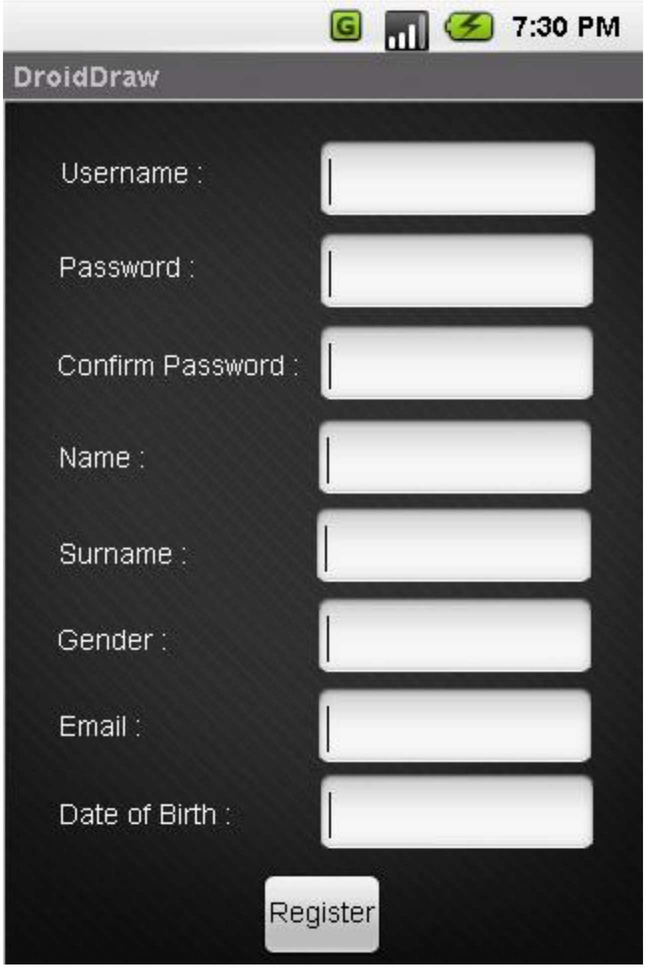


Figure 12 - Login Page

6.1.2. Register Page

In register page there are eight text boxes and one button. Text boxes are for entering username, password, confirm password, name, surname, mail address, gender and birth date. Button is register button which checks the validity of this information.



7:30 PM

DroidDraw

Username :

Password :

Confirm Password :

Name :

Surname :

Gender :

Email :

Date of Birth :

Figure 13 - Register Page

6.1.3. Profile Page

In profile page user information and owned albums are displayed. User information includes username, name, surname, mail address, and gender and birth date. Albums section consists of categories created by user. Each category tab includes all album information in that category and whether it is lent to another user or borrowed from another user or neither.

There are also three buttons on the page. If a user is in one of the created categories he/she can click the “insert album” button and the list of all owned albums is shown, from there user can select the albums he/she wants to move to selected category, send a message to another user by messages button and lastly search an album or a user by search button in this page. User can add a new album to his/her collection by add album button which is in menu. To create a new category “create a new category” button from the menu should be clicked.

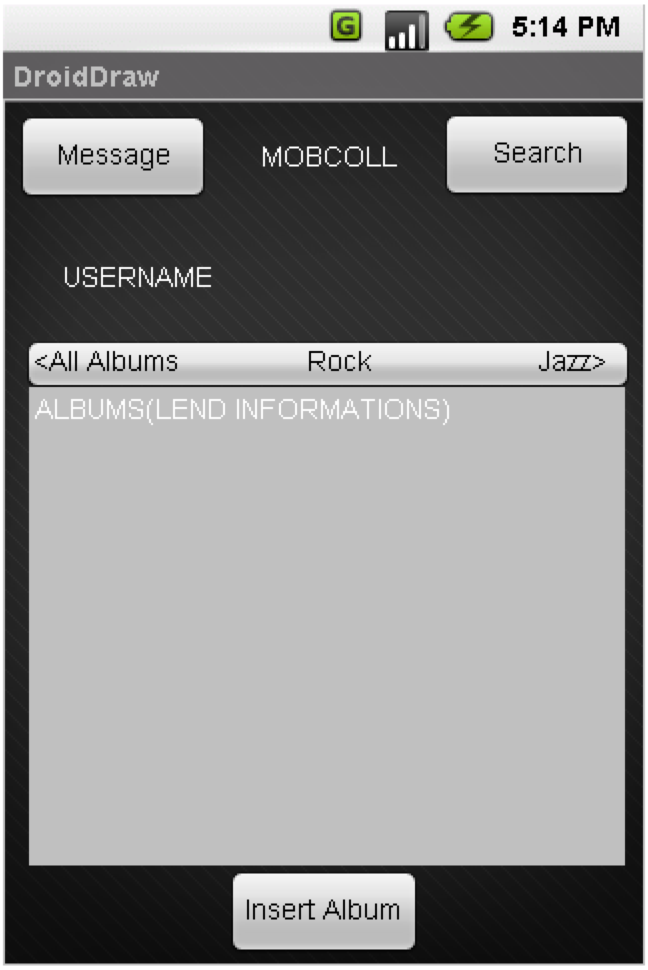


Figure 14 - Profile Page

6.1.4. Album Page

Album page consists of two main sections and a search button and add album button. First is album information section which includes album cover, name, barcode number, release year and its songs. This section also includes owners button. Clicking this button expands user name of owners of this album.

Second is comments section which includes the comments made about this album by all of the users. This section also includes a text box and add comment button.

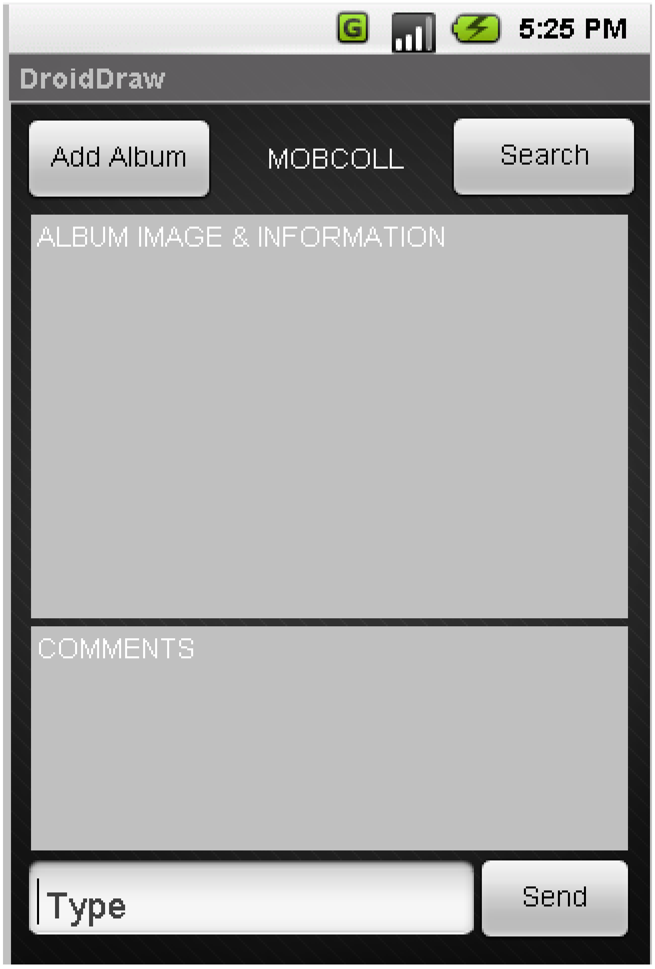


Figure 15 - Album Page

6.1.5. Message Page

Message page consists of three tabs. The first tab is inbox tab. In this tab user can see the messages that are sent by other users. The second tab is sent message tab. In this tab user can see the messages sent by user. The third and final tab is composing message tab. In this tab there are two text boxes and send button. Text boxes are for entering recipient's username and writing the message respectively.

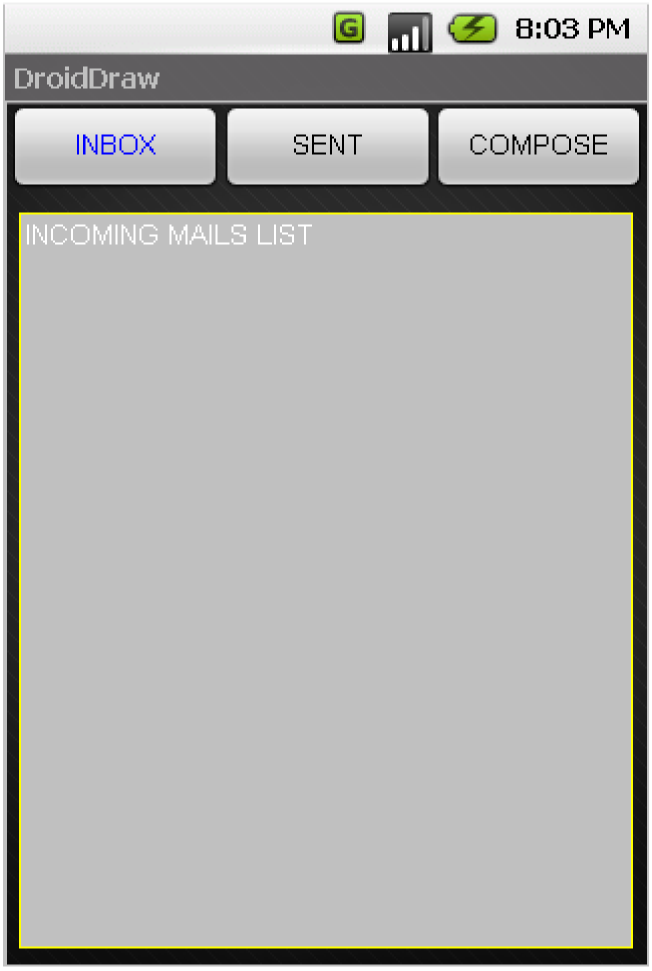


Figure 16 - Message Page Inbox

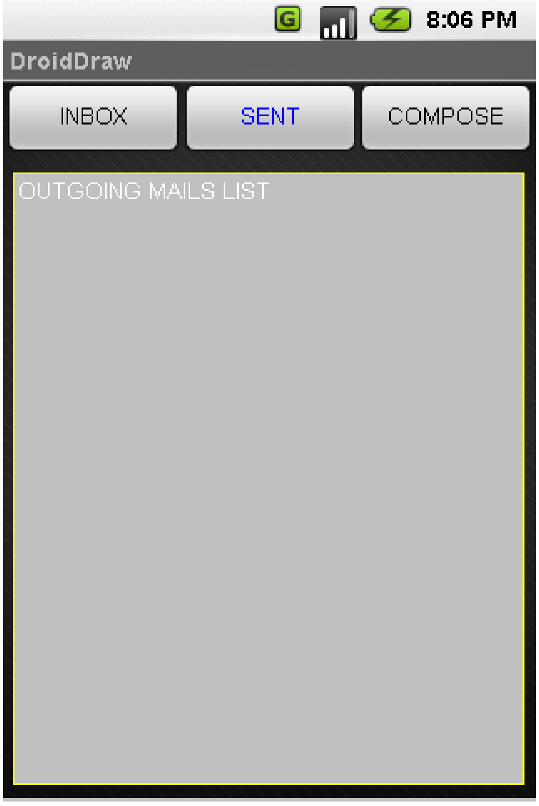


Figure 17 - Message Page Sent

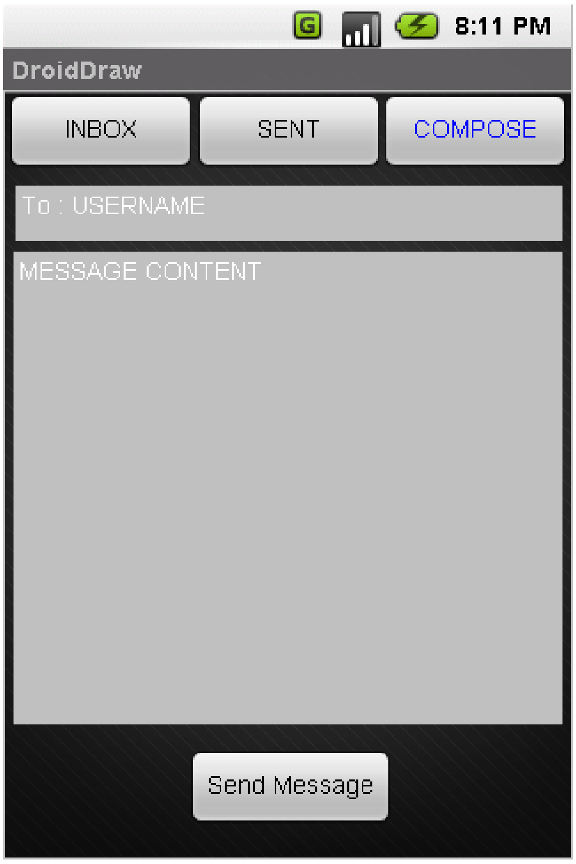
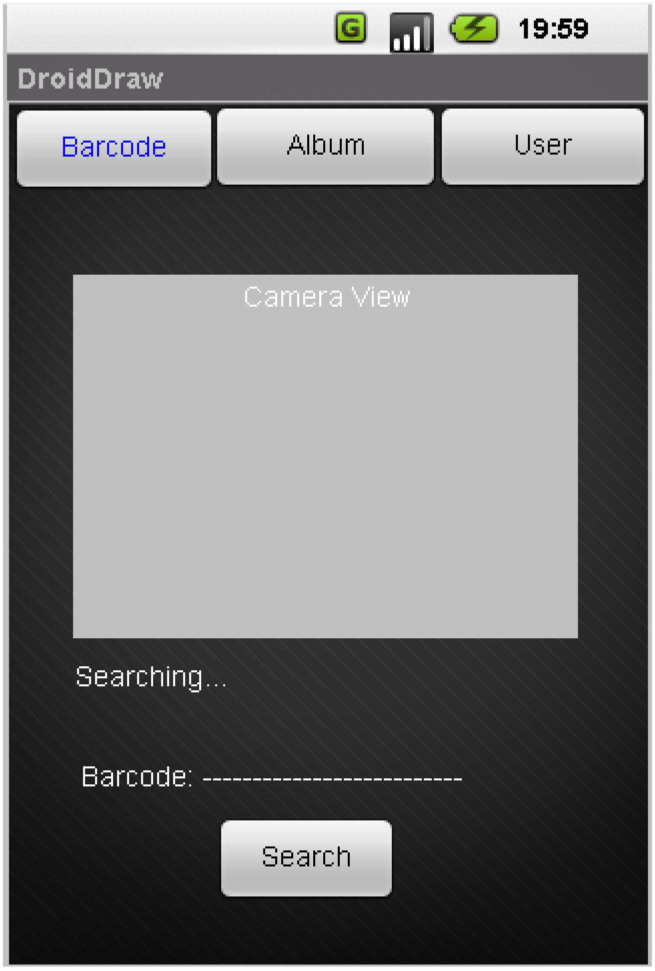


Figure 18 - Message Page Compose

6.1.6. Search Page

Search page consists of three tabs. The first tab is barcode search tab. In this tab mobile phone’s camera is run and a barcode can be scanned. If barcode is valid user is directed to the related album page. Second tab is album search tab. In this tab there are five text boxes and a search button. Text boxes are for entering album name, artist name, song name, release year and genre. The third and final tab is user search tab. In this tab there are three text boxes and a search button. Text boxes are for entering user name, name and surname.



Searching...

Barcode: -------------------------

Figure 19 - Search Page Barcode

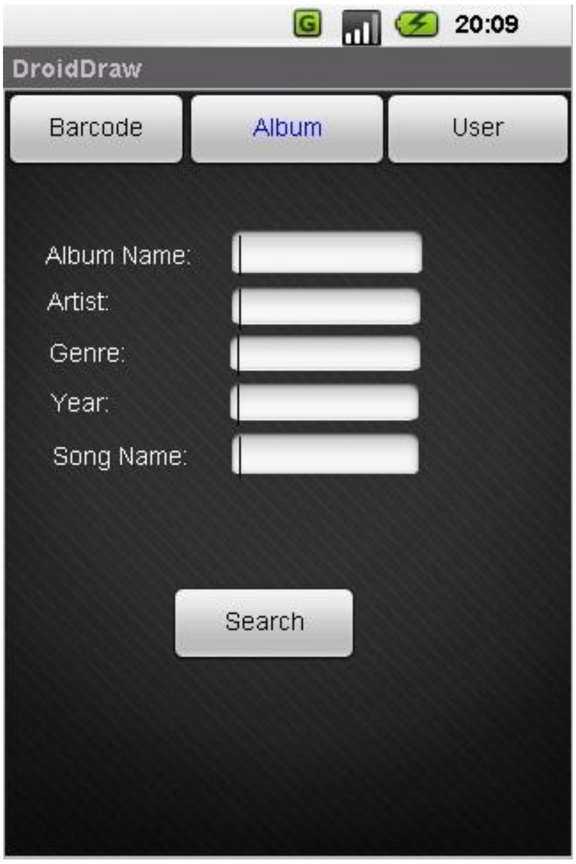


Figure 20 - Search Page Album

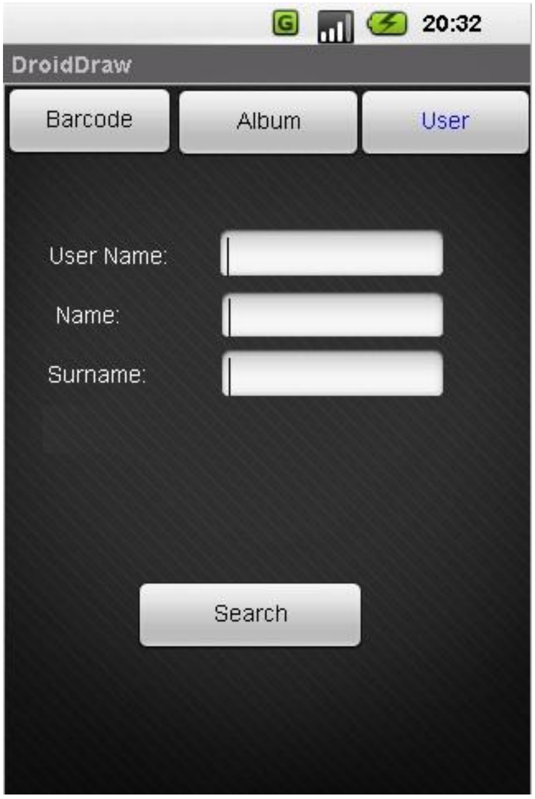


Figure 21 - Search Page User

6.2. Screen Objects and Actions

6.2.1. Login Page

- Login button: Checks if given username and password is valid. If it is valid directs user to profile page else stays in login page.
- Register button: Directs user to register page.
- Facebook login button: Sends Facebook login request.

6.2.2. Register Page

- Register button: Checks if given user information is valid. If it is valid directs user to login page else stays in register page.

6.2.3. Profile Page

- Add album button: Expands the album request section which includes album name and barcode.
- Message button: Directs user to message page.
- Search button: Directs user to search page.
- Create category button: Creates a new category.
- Insert album button: Inserts the selected album to selected category.

6.2.4. Album Page

- Search button: Directs user to search page.
- Add album button: Adds the album to users collection.
- Add comment button: Provides user to add a new comment about selected album.
- Owners button: Expands owners of the selected album section.

6.2.5. Message Page

- Inbox button: Opens the inbox tab of message page.
- Sent message button: Opens the sent message tab of message page.
- Compose message button: Opens the compose message tab of message page.
- Send message button: Send the written message to the selected user.

6.2.6. Search Page

- Barcode button: Opens the barcode search tab of search page.
- Album search button: Opens the album search tab of search page.
- User search button: Opens the user search tab of search page.

7. REQUIREMENTS MATRIX

	Server	Client	Database
Create Profile – 3.1	X	X	X
Login – 3.2	X	X	X
Logout – 3.3		X	
Update Album Section -3.5	X	X	X
Categorizing – 3.6	X	X	X
Search Album -3.7	X	X	X
Search User – 3.8	X	X	X
Glance Other User Profile -3.9	X	X	X
Send Message - 3.10	X	X	X
Add Comment – 3.11	X	X	X
Lend Album – 3.13	X	X	X