# SOFTWARE DESIGN DOCUMENT

**Version:** 1.1

**Date:** 22.12.2013

## MobileLibrary Project


**Prepared By:** HebeleGubeleGom Team

Ali Sahin

Ali Cinar

Yunus Emre Avci

Upol Ryskulova

**Preface**

This document contains the system design information for MobileLibrary project. The document is prepared according to the "IEEE Standard for Information Technology – Systems Design – Software Design Descriptions – IEEE Std 1016 – 1998".

This Software Design Documentation provides a complete description of all the system design and views of MobileLibrary Project.

The first section of this document includes purpose, scope, overview, reference material, definitions, and abbreviations of the project.

The second chapter of this document includes an overview of the functionality of the application. It describes the informal design generally.

The third chapter of this document will give the user a detailed description of each function of the system.

The fourth chapter of this document contains data design and data description of the project.

The fifth chapter of this document contains a general overview of what the user interface will look like.

The sixth and last chapter of this document includes of requirements matrix of the project.

## Change of History

| Version Number | DATE | Number of figure | A*,M*,D* | Title of Brief Description |
|---|---|---|---|---|
| 1.0 | 31.10.2013 | | | Original |
| 1.1 | 22.12.2013 | | M*,A* | Add new classes and viewpoints. |

**A* : Added**
**M*:Modified**
**D*:Deleted**

## Table of Contents

## Table of Figure:

# 1. Introduction

## 1.1 Scope

In this report, we are planning to give information for programmer to write a code. Therefore, the document includes needs of the programmer for code development.

## 1.2 Purpose

This document describes the conceptual design of the MobileLibrary Project according to the document guidelines presented in the IEEE 1016-1998 Recommended Practice for Software Design Descriptions (SDD).

The SDD shows how the software system will be structured to satisfy the requirements identified in the software requirements specification. It is a translation of requirements into a description of the software structure, software components, interfaces and data necessary for the implementation phase. In essence, the SDD becomes a detailed blueprint for the implementation activity. In a complete SDD, each requirement must be traceable to one or more design entities.

## 1.3 Overview

The purpose of this document is to help the reader visualize the solution to the project presented. This document verifies how the design meets the requirements stipulated in the SRS document through design viewpoints. The design viewpoints will cover all design elements presented before.

By using information from IEEE 1016-1998, this document will provide a direct approach to the development of this project hence reducing feature creep and pointedly determine the quality of the design.

## 1.4 Reference Material

IEEE, *IEEE Std 1016-1998 Recommended Practice for Software Design Descriptions*, 1998-09-23, The Institute of Electrical and Electronics Engineers, Inc., (IEEE )
IEEE, *IEEE 1016* Software Design Document (SDD) Template for CENG491

## 1.5 Definitions and Abbreviations

| **Term** | **Definition** |
|---|---|
| Database | Collection of all the information monitored by this system |
| User | METU Library user |
| Android | A mobile device operating system developed by Google Inc. |

| | |
|---|---|
| ISBN | The International Standard Book Number (ISBN) is a unique numeric commercial book identifier. |
| IEEE | The Institute of Electrical and Electronics Engineers (IEEE) is a professional association headquartered in New York City that is dedicated to advancing technological innovation and excellence. |
| Software Requirements Specification | A document that completely describes all of the functions of a proposed system and the constraints under which it must operate. For example, this document |
| Software Design Description | A document that completely describes all of the function of a proposed system and the constraints under which it must operate. |
| JSON | A lightweight data-interchange format(JavaScript Object Notation) |
| Eclipse IDE | A multi-language software development environment |
| RESTFUL | An approach for getting information content from a Web site by reading a designed Web page that contains an XML. |

# 2. Conceptual model for software design descriptions

Information about concepts and context of SDD , the stakeholders will be given in this part

## 2.1 Software design in context

### 2.1.1 Technologies Used

The system is coded with Java programming language by using Eclipse integrated development environment. We will use MySQL for database. Android SDK will be used for android application development and JSON will be used for RESTFUL web service.

### 2.1.2 Application Overview

The main goal of the project is to enable user to create a virtual library owned by him/her. The final product of Mobile Library project will be a mobile platform for Android mobile phones or tablet systems that will enable third party mobile application developers to easily develop Android based collection applications by utilizing the common Android services. Adobe Flash CS6 is used for the animation which shows the locations of the materials in the METU Library.

The Mobile Library project will be an android application that is designed for the METU Library users. By using their METU accounts, users can be logged in to the system and they can use all the services provided by Mobile Library application. For example, users can search the materials in the METU Library, view the locations of the METU Library materials, pay their debt for the METU Library, view the announcements of the METU Library, make suggestions for the METU Library, etc. Users also can use some of these services without logging.

## 2.2 Software design descriptions within the life cycle

### 2.2.1 Influences on SDD preparation

System Requirements Specifications is the main sources for SDD preparation. The functional and non-functional requirements are main factors to determine the design of the project.

### 2.2.2 Influences on software life cycle products

In the process of implementation of the project and preparation of the SDD, Some requirements can be changed due to unexpected constraints.

Testing part of the project can be prepared by using the information in System Design Documents.

#### 2.2.3 Design verification and design role in validation

Verification is the determination whether a software work product fulfills specified requirements. Validation is the determination that the requirements for a specific intended use of a software work product are fulfilled. Therefore, verification and validation results are controlled by the requirements.

## 3. Design description information content

## 3.1 SDD identification

At the end of the second semester, all METU students or guests who have android platform can use project “Mobile Library”. Also, we are planning to prepare simple interface which is simple and user-friendly. This project will reduce the workload of the library personals because of the barcode reader which has not been used by other libraries earlier.

## 3.2 Design stakeholders and their concerns

In this project, METU official accounts and passwords are used. Therefore, the stakeholders’ main concern is security and efficiency.

## 3.3 Design views

Unified Modeling Language (UML) 5.02 is used for graphical representations of viewpoints in “MobileLibrary” Project in Diagrams. ADK is used for user interface design in sixth section of this project.

## 3.4 Design viewpoints

Interface, Logical, information, context and use case are design viewpoints which we provide information about in this document.

## 3.5 Design rationale

We mostly focus on sustainability and efficiency features because it is important for us that the application can be used all kind of METU students and METU Library guests. Efficiency is important because there is huge number of applications and we want our project to be preferable. While writing code, we comment all methods in order to be read by the stakeholders who want to integrate the project.

## 3.6 Design Languages

Unified Modeling Language (UML) 5.02 is used for graphical representations of viewpoints in Mobile Library Project in 3.System Architecture, 4.Data Design and 5.Component Design parts. ADK is used for user interface design in sixth section of this project.

# 4. Design Viewpoint

## 4.1 Logical viewpoint

This viewpoint aims to show the key abstractions such as classes and interactions among them. UML Class diagram is provided for this aim which can be seen as below;

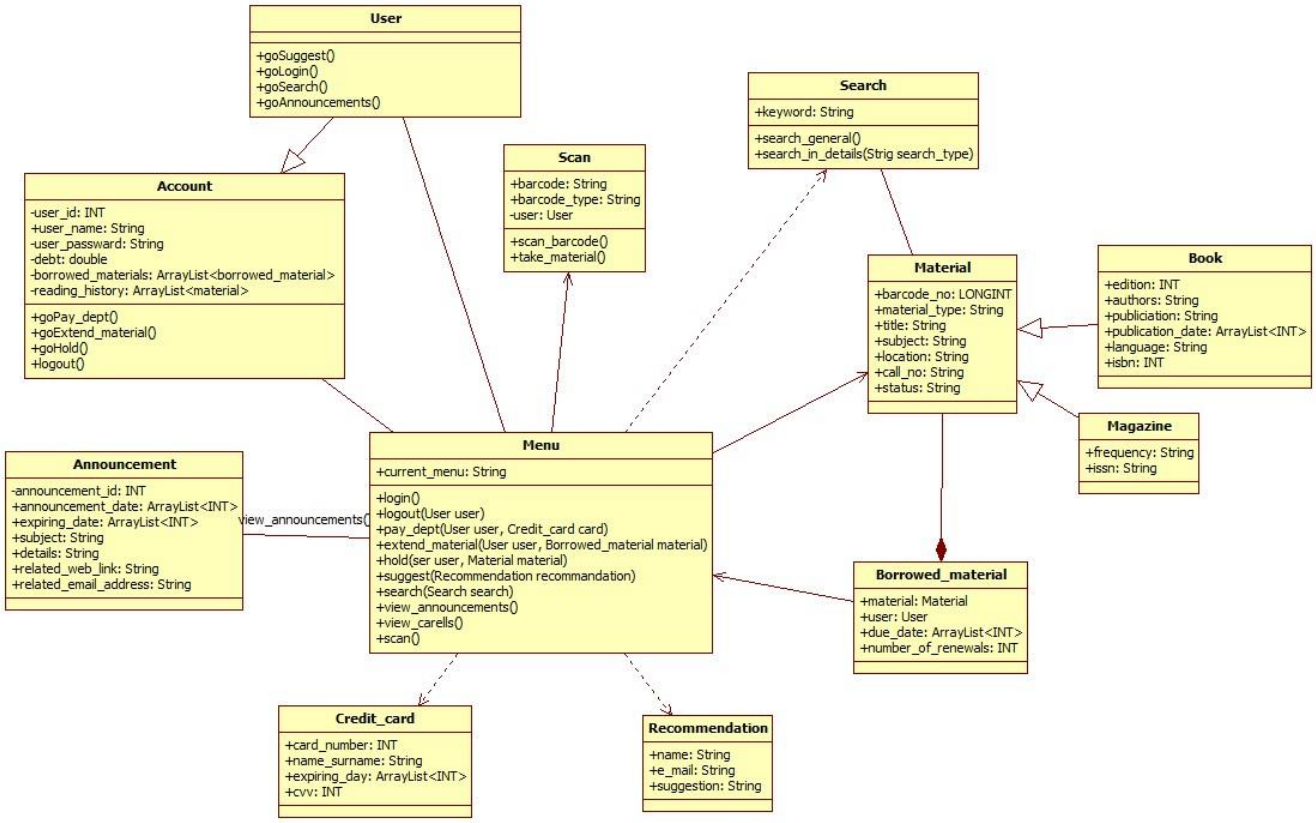


**Figure 1 – Class Diagram of MobileLibrary**

## 4.1.1 Design concerns

The Logical viewpoint is used to address the development and reuse of adequate abstractions and their implementations.

## 4.1.2 Design elements

**account:**

**user_name:** Name of the account taken from the user.

**user_passward:** Password of the account taken from the user.

**user_id:** ID number of the account taken from the user.

**reading History:** List of the materials that user of the account borrowed up to date.

**messages:** Massages received from the Library.

**debt:** Debt of the user of the account because of late giving back borrowed materials.

**material:**

**material Type:** Types of the materials that can be borrowed in the Library. There are two types of materials; book and magazine.

**barcode NO:** Barcode number of the material. This property is unique for any material.

**title:** Title of the material.

**subject:** Subject terms of the material.

**location:** Location of the materials. Shows the book is in the main library or in the North Cyprus Library.

**call_no:** Call Number of the materials. This number used for the location of the book in the library.

**status:** Status of the materials. Shows whether the materials is available in the library or not.

**book:**

**edition:** Edition of the book.

**authors:** Authors of the book.

**publication:** Publication information of the book.

**Publication_date:** Publication date of the book.

**language:** Language that book has been written in.

**isbn:** ISBN number of the book.

**magazine:**

**frequency:** Publishing frequency of the magazine.

**issn:** ISSN of the magazine.

**borrowed_material:**

**material:** The material. This class includes the properties of the material.

**borrower:** The user who has borrowed the material.

**due_date:** Due date of the material.

**number_of_renewal:** Number of renews of the material. Users can renew the borrowed material at most three times.

**announcement:**

**announcement_date:** Date when the announcement is announced first.

**expire_date:** Date when the announcement will be unpublished.

**subject:** Subject of the announcement.

**details:** Details of the announcement.

**related_web_link:** Web link that is related with the announcement.

**related_email_address:** E-mail address that is related with the announcement.

**credit_card:**

**card_number:** Card number of the credit card.

**name_surname:** Name and Surname of the credit card owner.

**expiring_day:** Expiring date of the credit card.

**cvv:** CVV number of the credit card.

**recommendation:**

**name:** Name of the person who makes the recommendation.

**e_mail:** E-mail address of the person who maked the recommandation.

**suggestion:** The suggestion message.

**menu:**

**current_menu:** The menu that the user currently resides in.

**search:**

**keyword:** The keyword that is going to be searched.

**search_type:** The type of the search. It can be detailed search or quick search.

**Scan:**

**Barcode:** The keyword that is read by camera

**Barcode_type:** It is the string that holds type of the barcode

**User**: In which user account the barcode is read

## 4.2 Information viewpoint

An Entity Relationship Diagram is shown as below;

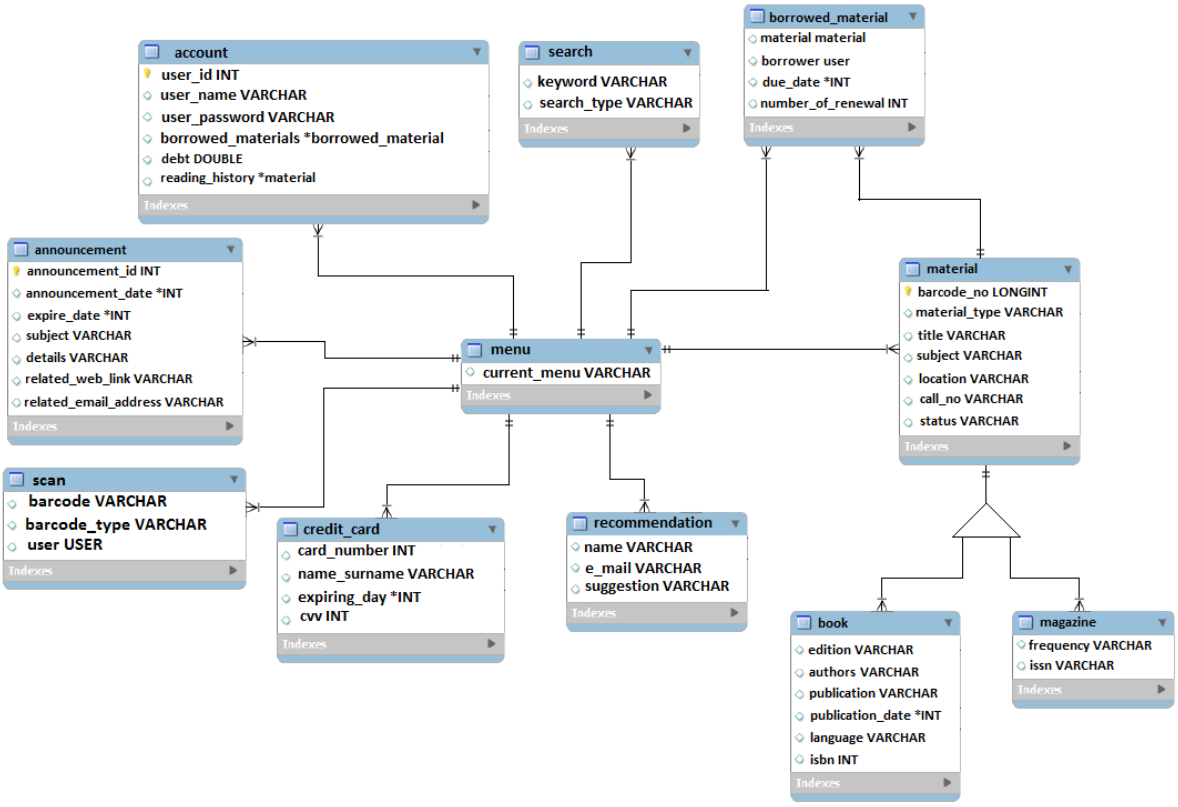


**Figure 2 – ER Diagram**

### 4.2.1 Design Concern

This viewpoint aims to show the way that the system stores, manages, manipulates, the persistent information that the system will maintain.

### 4.2.2 Design Elements

Alphabetically list the system entities or major data along with their types

**1. Account:**

| **Parameter:** | **Parameter Type:** |
|---|---|
| **borrowed_materials:** | List of borrowed_material |
| **debt:** | Double |
| **reading_histort:** | List of material |
| **user_id:** | Integer |
| **user_name:** | String |

| | |
|---|---|
| **user_password:** | String |

**2.Announcement:**

| **Parameter:** | **Parameter Type:** |
|---|---|
| **announcement_date:** | Date |
| **details:** | String |
| **expire_date:** | Date |
| **related_email_address:** | String |
| **related_web_link:** | String |
| **subject:** | String |

**3.Book:**

| **Parameter:** | **Parameter Type:** |
|---|---|
| **authors:** | String |
| **edition:** | String |
| **isbn:** | Integer |
| **language:** | String |
| **publication:** | String |
| **Publication_date:** | Date |

**4.Borrowed_material:**

| **Parameter:** | **Parameter Type:** |
|---|---|
| **borrower:** | Account |
| **due_date:** | Date |
| **material:** | Material |
| **number_of_renewal:** | Integer |

**5.Credit_card:**

| **Parameter:** | **Parameter Type:** |
|---|---|
| **card_number:** | Integer |
| **cvv:** | Integer |
| **expiring_day:** | Date |
| **name_surname:** | String |

**6.Scan:**

| **Parameter:** | **Parameter Type:** |
|---|---|

| | |
|---|---|
| **barcode:** | String |
| **barcode_type:** | String |
| **user:** | User |

## 4.3 Interface viewpoint

This section briefly describes the interface and components of MOBILE LIBRARY. In this viewpoint, there will be two different interfaces which are user-application interface and application-databases. Application and databases transforms the information about users and books via Curl and Json objects by using php functions.

### 4.3.1 Design Concern

Concerns of the stakeholders in this document are simplicity and efficiency.

### 4.3.2 Design Elements

#### 4.3.2.1 Starting Screen

The starting screen will be the first screen which users will see when MOBILE LIBRARAY application starts. It contains info about the library hours, login information and menu buttons.

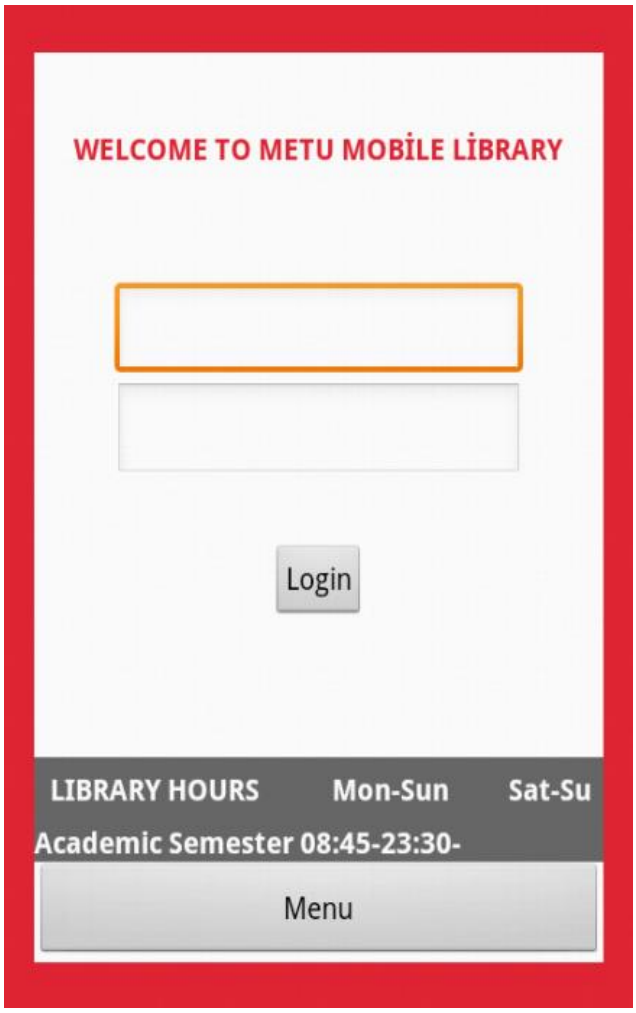


**Figure 3 - Starting Screen of MobileLibrary Application**

### 4.3.2.2 Logout Screen

When the logout button is clicked, the start screen will show up again. After that, the system will be on idle mode, waiting for another logging action or another menu activity.

### 4.3.2.3 User Profile Screen

When a user logs in, the screen below will show up. On this screen, there will be several fields containing some information about the user. The user will also have the opportunity to view his/her payment record, to see his/her recommended book list and read announcements. He/she can also make an advanced search on the library database. These items will be placed on the screen as shown in the image below.

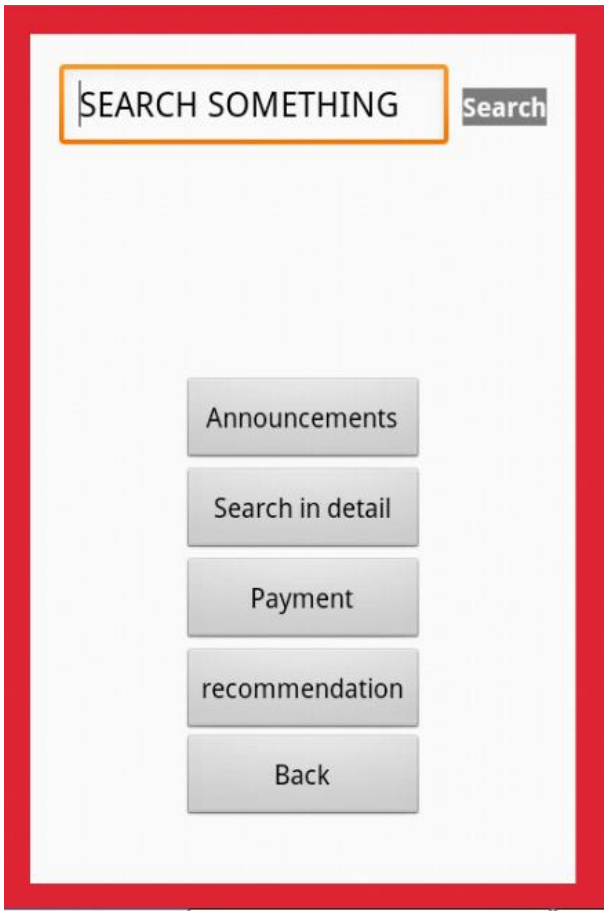


**Figure 4 - User Profile Screen of MobileLibrary Application**

### 4.3.2.4 Announcements Screen

This screen will contain a list of announcements published by the Library management. The list will contain info about events organized by the library. These events could be library orientation days, library tour days, excel or library database search workshops, and other similar activities. Users also can find information about holiday operation hours, website maintenance or other potential last minute updates. Please see the image below.

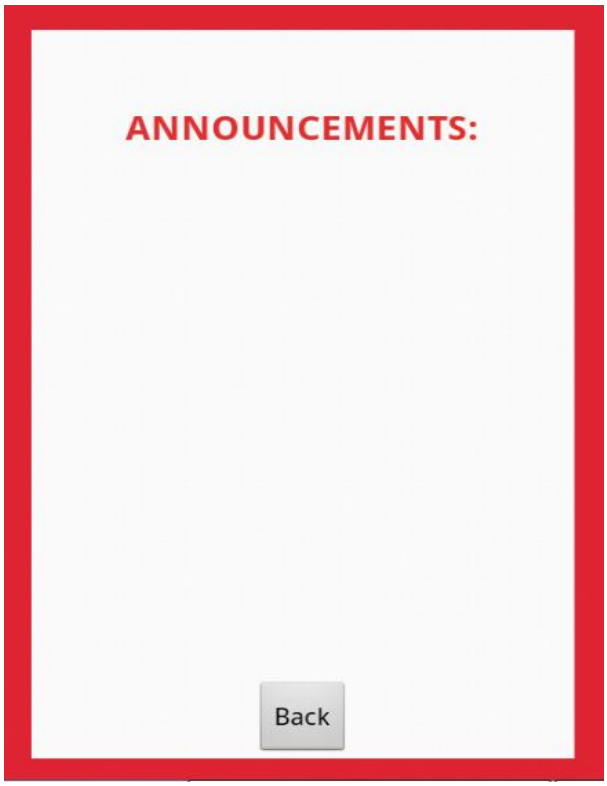


**Figure 5 – Announcements Screen of Mobile Library Application**

### 4.3.2.5 Library Catalogue Search Screen

This library catalogue search screen consists of an advanced database search operation utility. A combo box that displays specific information about searched books will be above the list. This combo box will contain information such as title, author, subject and keyword choices. At the start of each database search activity, the user could perform his/her search based on title, author, subject or a specific keyword. These items will be located on the screen as shown in the following image.

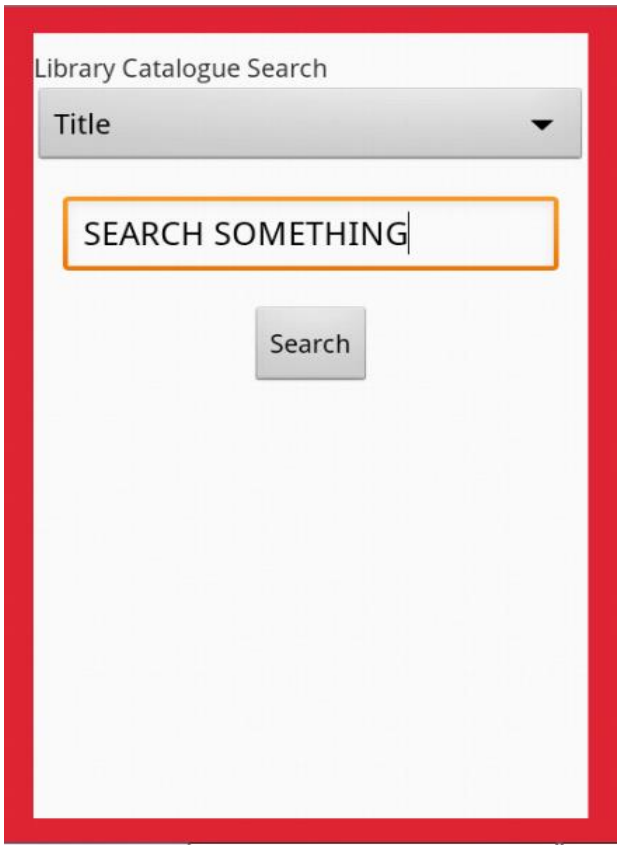


**Figure 6 – Library Catalogue Search Screen of MobileLibrary Application**

### 4.3.2.6 Payment Screen

A user can pay his/her library dues using this screen. The screen contains information that is necessary for the user to properly make the payment. These are three edit text fields that the user can enter his/her credit card information: “Card Number”, “Expiration Date” and “CVV”. The “Approve” button is also located on the screen for the user to confirm that whether the information provided correctly. These items are located on the screen as shown on the following image.

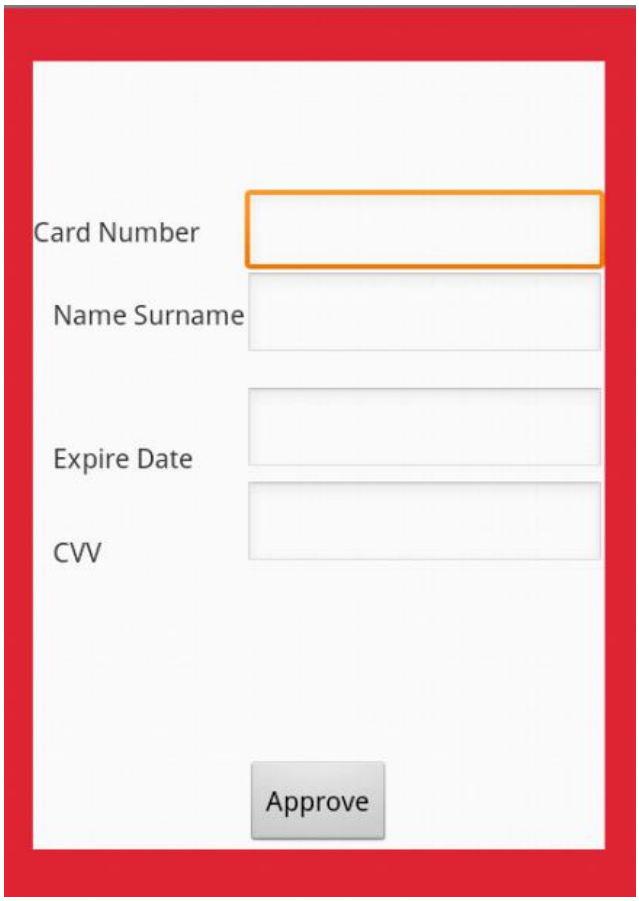


**Figure 7 – Payment Screen of Mobile Library Application**

### 4.3.2.7 Recommendation Screen

This screen will make it easier for users to suggest materials that they want to see in their library. They can also submit their complaints and the recommendations directly to the library management. This screen will enable the library management to respond very fast to the issues that users face when they use library services. Please see the image below as an example.

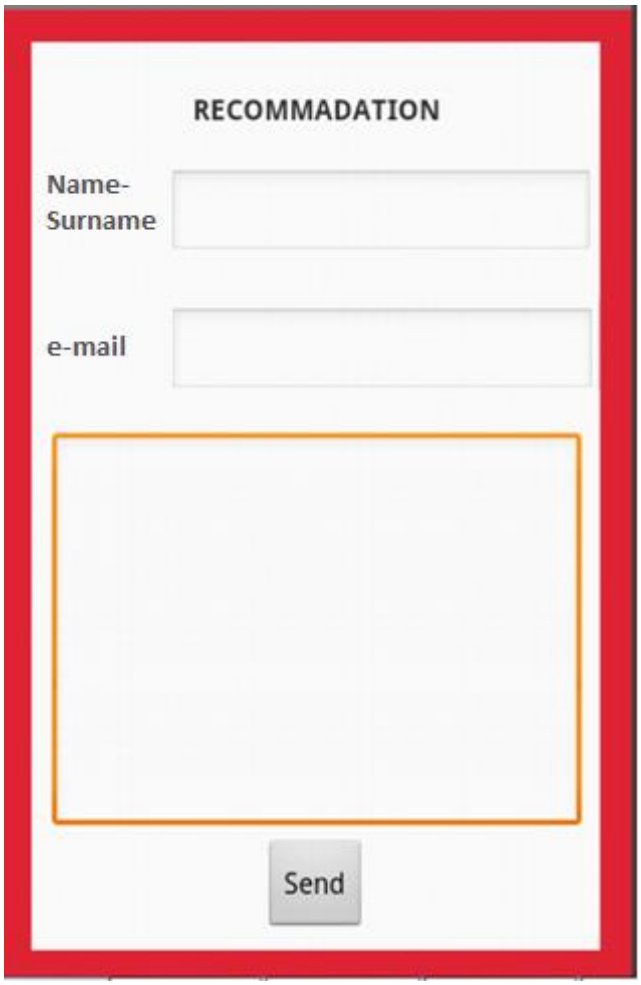


**Figure 8 – Recommendation Screen of Mobile Library Application**

### 4.3.2.8 Book Screen

This screen will make it easier for users to get information about books that they have selected. They can also see book' location, call number, barcode number, status. There will be also a simulation button to help users to find book in the library easily. If user presses the simulation button then a simulation will appear. In this simulation, the shelf of the material will be winked constantly. Please see the image below as an example for book and simulation.

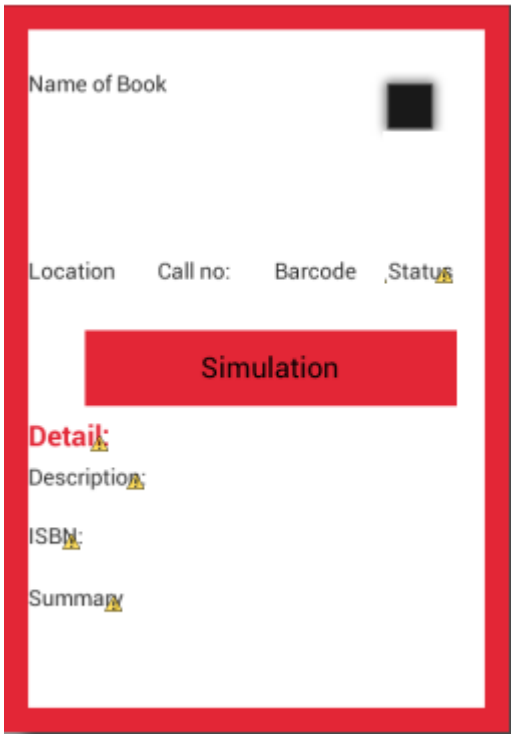


**Figure 9 – Book Screen of MobileLibrary Application**

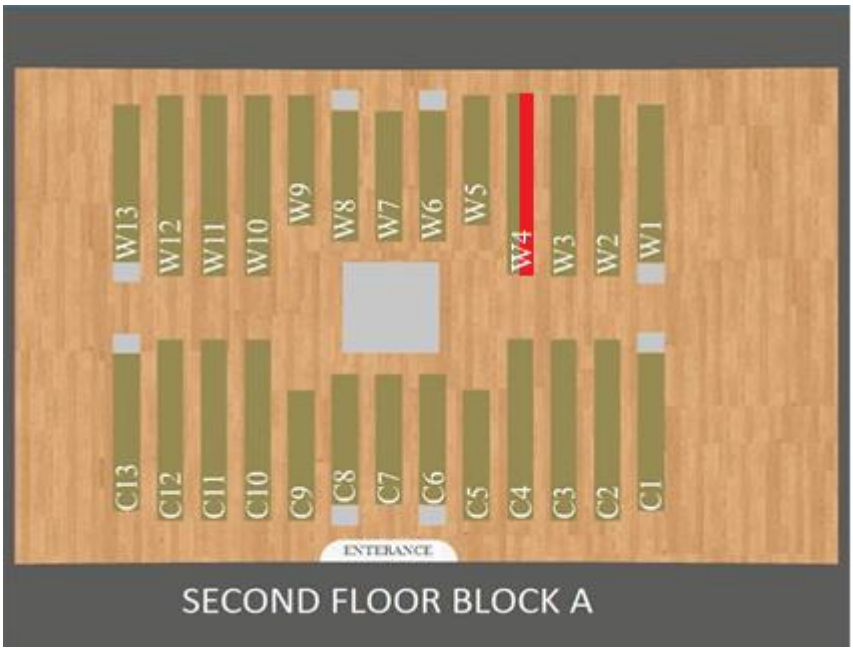


**Figure 10 – Simulation Screen of MobileLibrary Application**

### 4.3.2.9 Other Interfaces

Database connection is provided by php functions which transport the information between application and databases. Since databases of the library are connected via Millennium Company which is the company in USA, we need the php functions which take the information and convert them into curl or json objects and send us.

The connection between these components will be carried on internet with HTTP protocol.

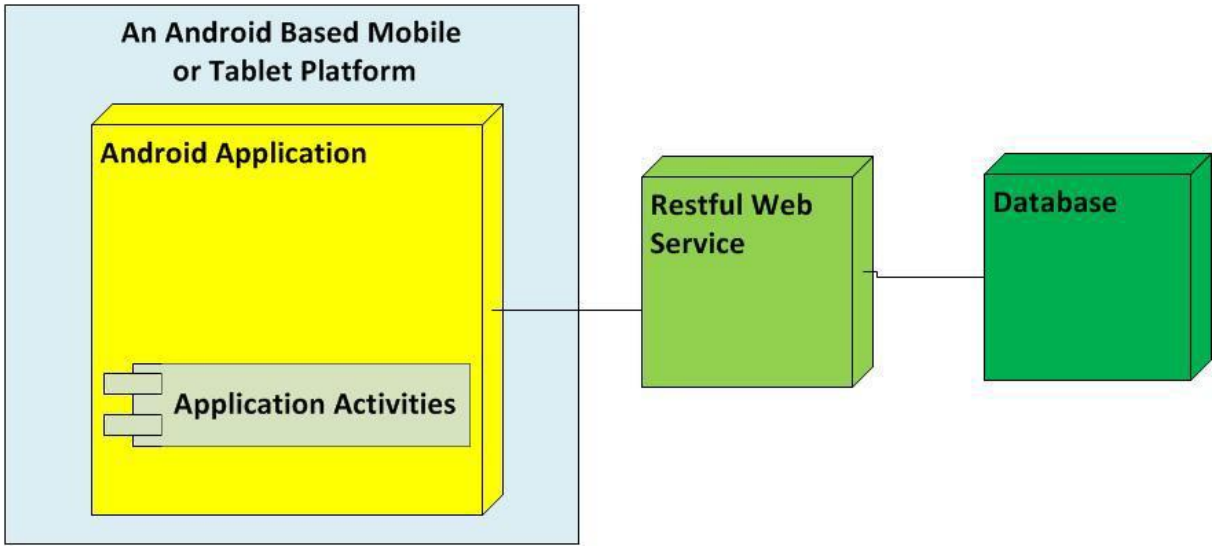


**Figure 11 - General Deployment Diagram**

In the picture, Restful Web Service is the service which stores the php codes and provides the database content to users.

## 4.4 Interaction viewpoint

### 4.4.1 Login Component

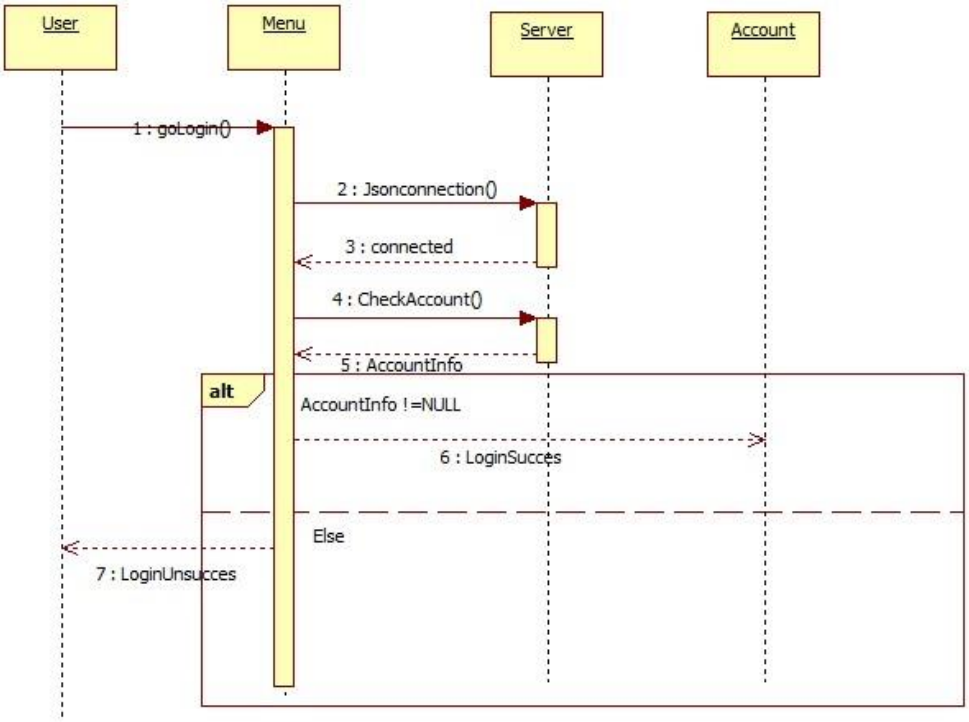


**Figure 12 – Sequence Diagram of Login Component**

In Mobile Library Application, data exchange between Server and android application for login is achieved for JSON. There are three different classes used by login event. For login operation, METU username and password are needed. Firstly, the connection with remote server is set via JSON. Secondly, username and password are checked and *CheckAccount* returns the result. If the AccountInfo is NULL, then the program gives a warning like that account information is wrong. Otherwise, *checkAccount()* will return an account object of the User.

### 4.4.2 Search Component

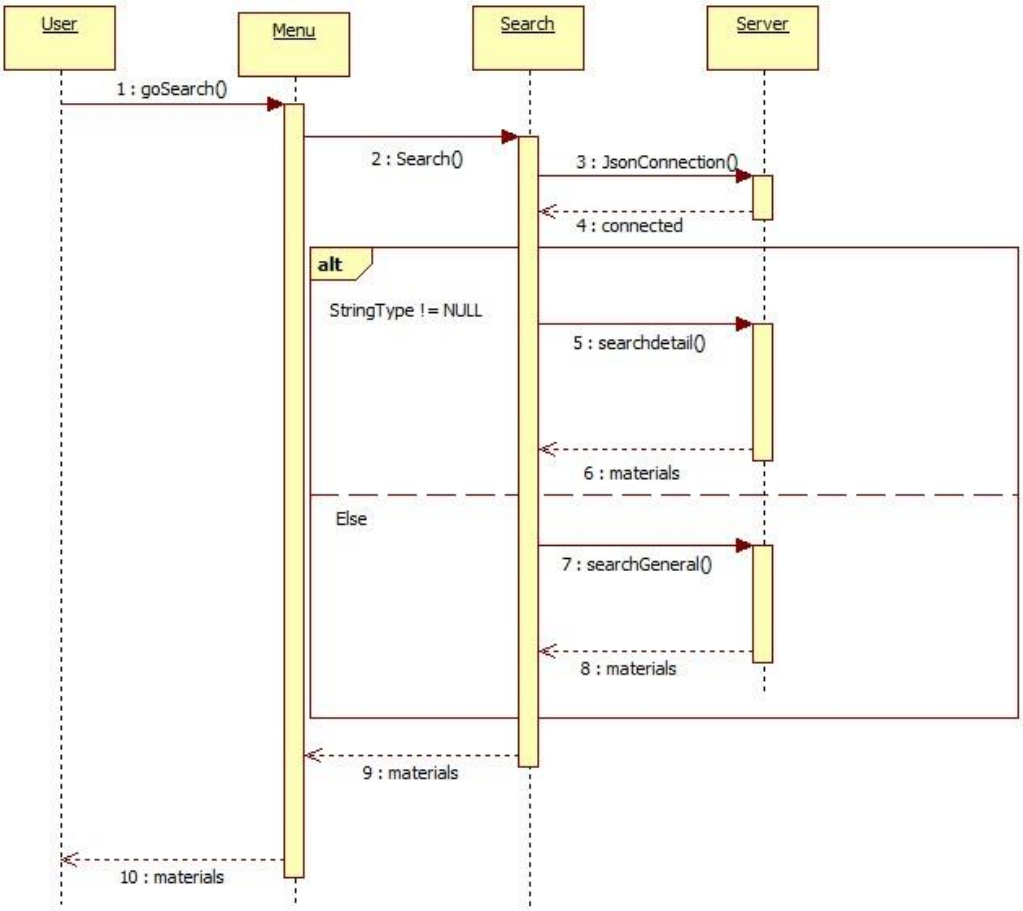


**Figure 13 – Sequence Diagram of Search Component**

In Mobile Library application, user can search all materials in the METU library by writing only keyword in general or by giving detail type and keyword. Three different classes are used in this operation which are user, menu and search. Keyword is needed for search. Firstly, the application connects the JSON. If the connection is hold, the application checks the searching inputs. If the searching type is NULL, *search_general* is called and results are taken as a material array. Otherwise, the *search_in_details* is called and returns material array. At the end of the function search, the operation displays the search results. This operation can be used by user even if user does not login the system.

### 4.4.3 Suggest Component

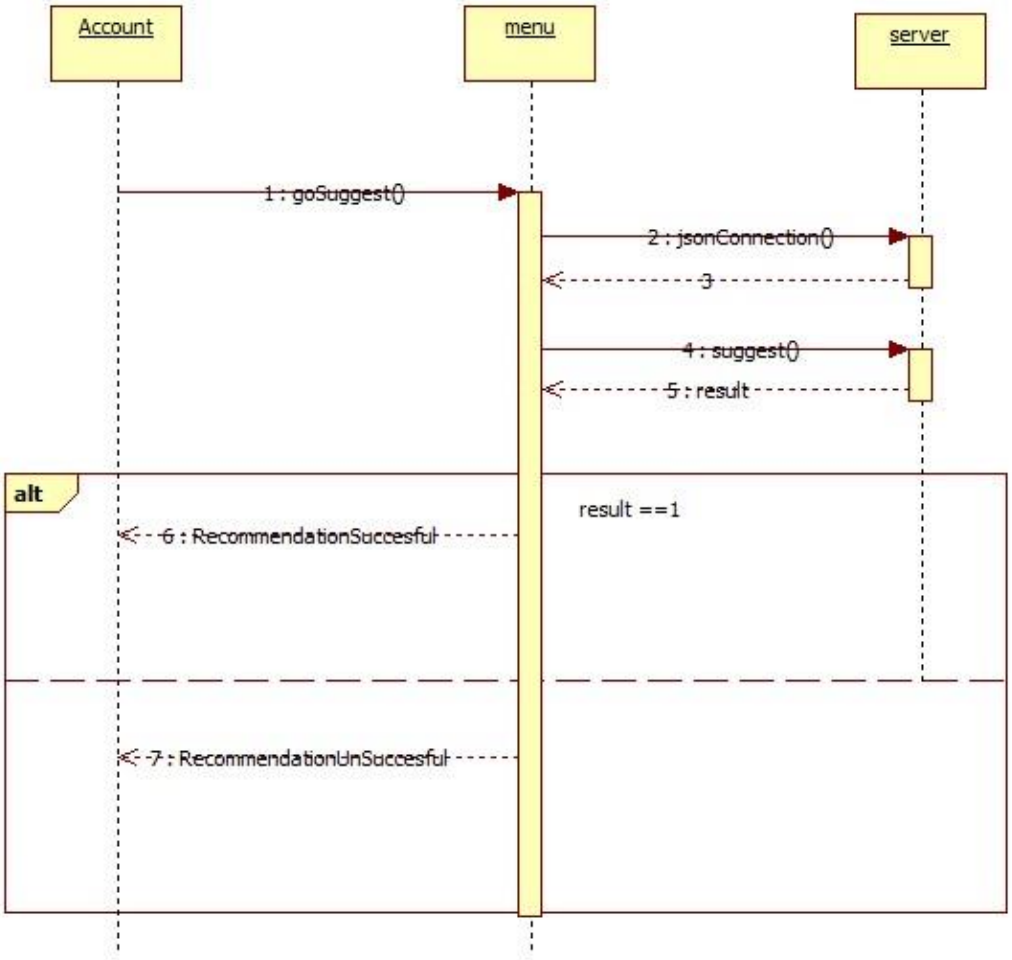


**Figure 14 – Sequence Diagram of Suggest Component**

In the Mobile Library application, the user which has already logined the system can suggest new materials in order that the library staffs can buy them. Firstly, the application checks the JSON connection. If the connection is hold, new recommendation class sends the server in order to be saved. At the end of the operation, result is checked in order to understand whether the recommendation saves in server successfully or not and finally, user is informed. Recommendation class has the name, e-mail and suggestion variables which are taken as a input from the user.

### 4.4.4 View Announcement Component

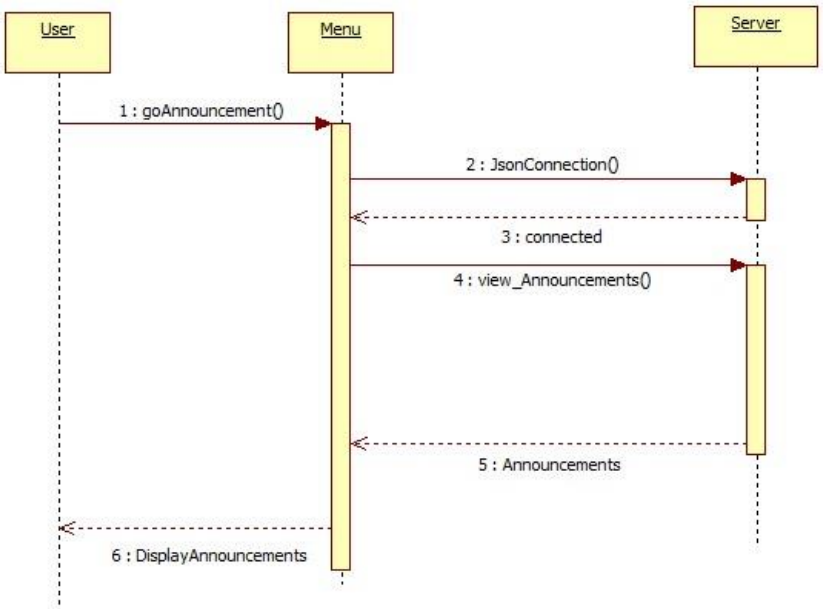


**Figure 15 – Sequence Diagram of View Announcement Component**

In Mobile Library application, the user can view the announcements which are determined by administrations of the METU library. Firstly, the user clicks the announcement button and *view_Announcement* button is called. The *view_Announcement* methods firstly check the JSON connection and then get recent announcements from Server. At the end of the methods, it displays the announcements on the screen. The user can examine the announcement whether the user logins or not. Also, the announcement classes have seven variables which are announcement_id, announcement_date, expiring_date, subject,details, related_web_link and related_email_address.

### 4.4.5 Extend Material Component

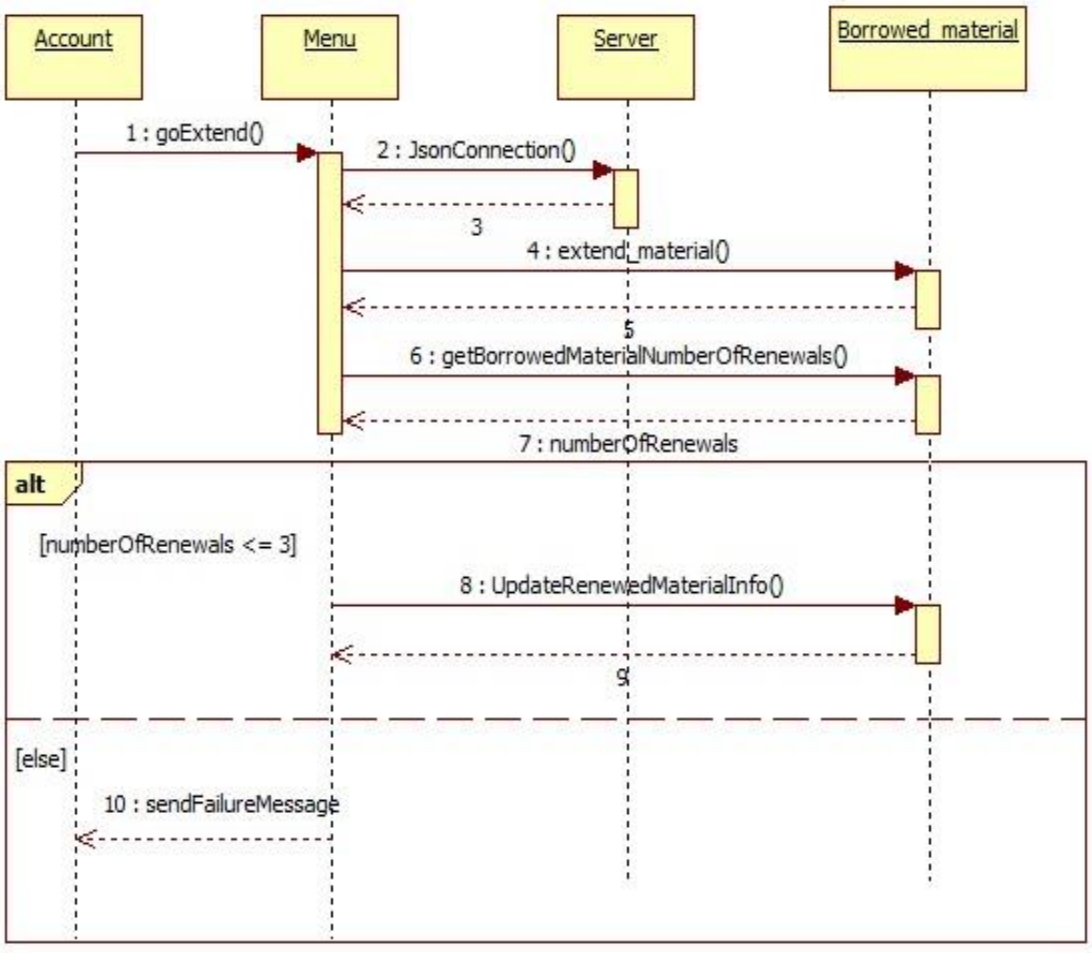


**Figure 16 – Sequence Diagram of Extend Material Component**

In the MobileLibrary application, the user which has already logged in the system can hold the book which is not available. We skipped the login part here since it is shown above in Login Component diagram. By user press to button “hold”, application will get that button press and calls goExtend() method. Next step of application is to make JSON connection successfully. In order to extend material first application checks whether the number of renewals is not more than 3, if this restriction holds then it updates the due date of material and increments the number of renewals by one. Otherwise, it will send the user a failure message stating that user is not allowed extend more than three times the borrowed material.

### 4.4.6 Hold Component

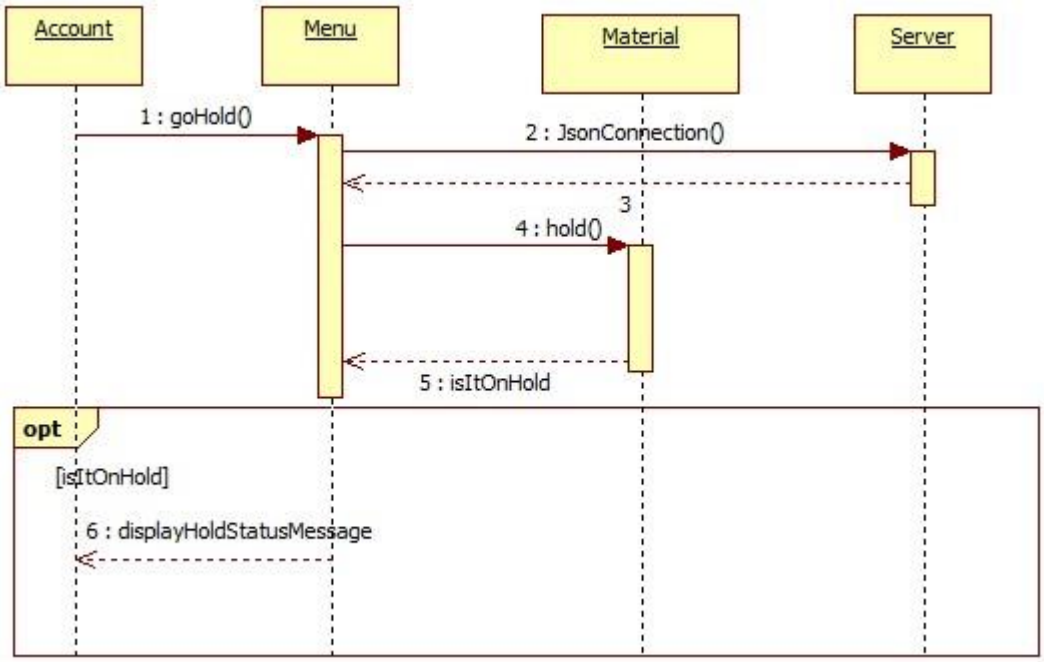


**Figure 17 – Sequence Diagram of Hold Component**

This diagram shows how holding a book done, after the user login to the system. Application first obtains connection through JSON. Then it goes to Material class to make a hold on that material. In Material class it tries to hold the material and send the isItOnHold with return type Boolean which indicates whether the material successfully held or not. Finally, Account class will display the message about success or failure.

### 4.4.7 Pay Debt Component Diagram

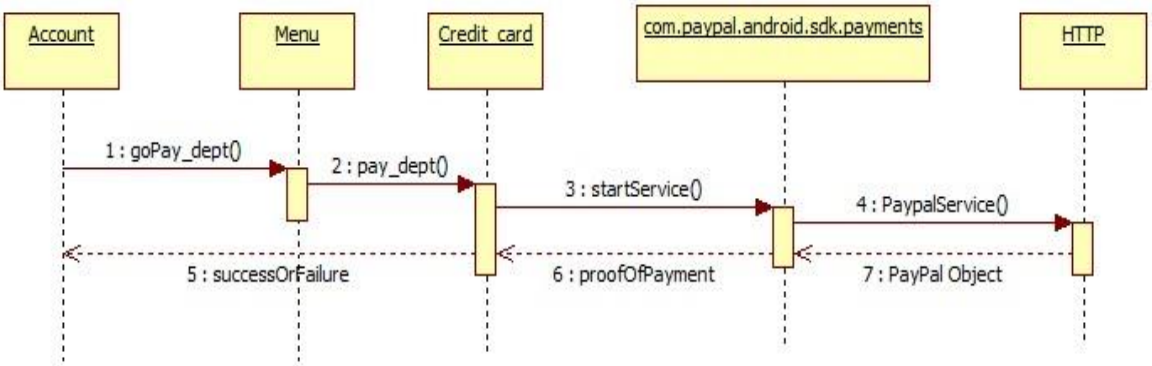


**Figure 18 – Sequence Diagram of Pay Debt Component**

In MobileLibrary Application users can pay his or her debt, debt which is penalized by METU Library for late submitting books or for not extending due date of a taken book, by one restriction that only the users with login success can use this feature of MobileLibrary application. Since the Login Component is shown above, here we skipped that part. We assume in this Pay Debt Component diagram that user login successfully. When the button is pressed our application will directly perceive this event as credit card payment which is supported by PayPal API. First it will pass payment information to create a payment and then start the service. Since the PayPal API uses HTTP it calls PayPalService() to start the transaction . It will return a PayPal Object which send payment approval. As a last step application will send a message stating that payment was successful or not. Here we are not delving deep into PayPal since it provides various payment related operations we skipped it for simplicity.

### 4.4.8 Logout Component

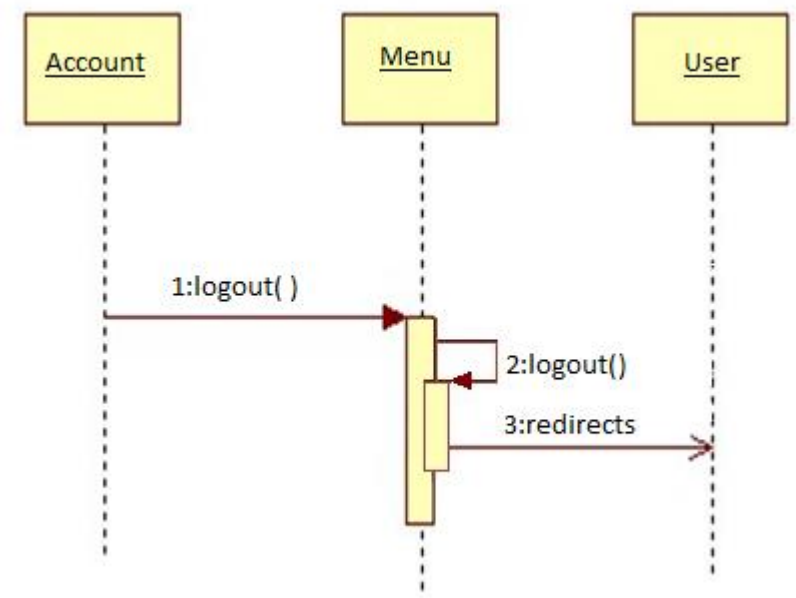


**Figure 19 – Sequence Diagram of Logout Component**

This sequence diagram shows logging out. When the user presses button “logout” it redirects user to the User class which means user find himself in the login page.

## 4.5. Context Viewpoint

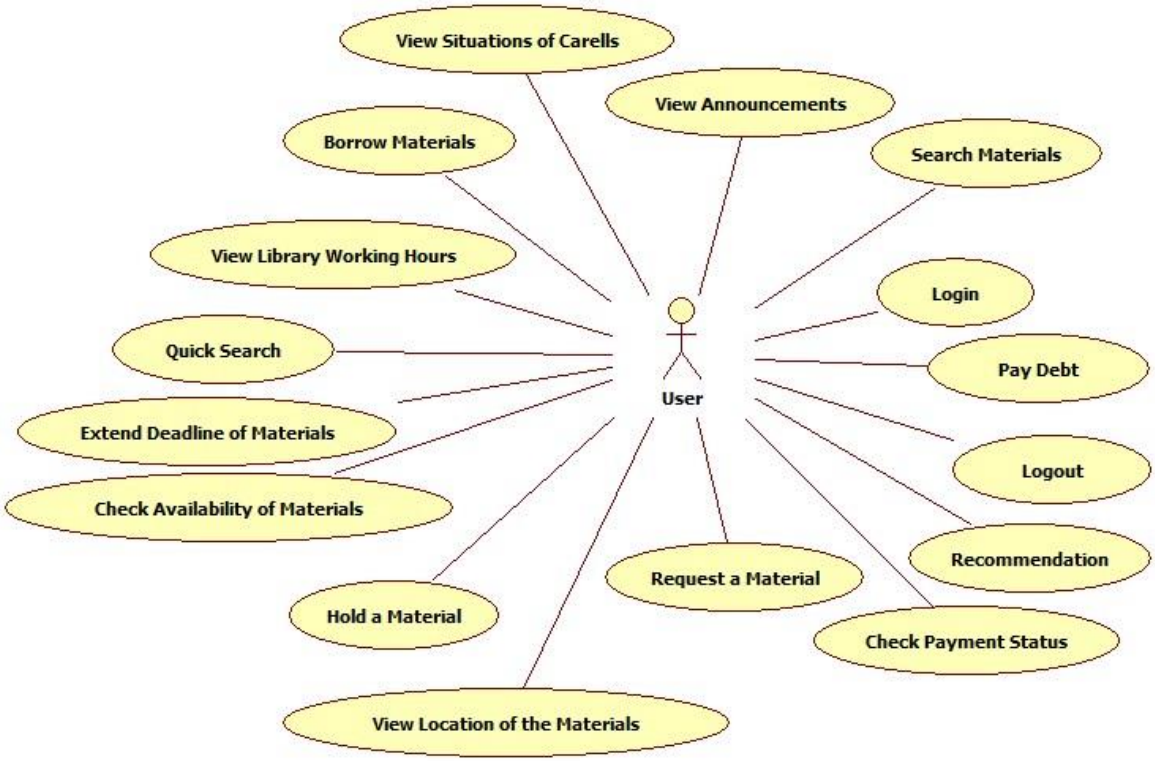


**Figure 20 – Use-Case Diagram of User**

### 4.5.1 Design Concern

Concern of the context viewpoint is to determine properties of the user. In terms of the efficiency of the user features, it is important to give explicit definition to these features.

### 4.5.2 Design Elements

Since almost all features are declared in System Requirement Specification documents, only new features are explained here

#### 4.5.2.1 Use Case: Borrow materials

**Diagram:**

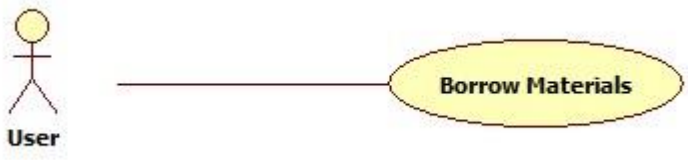


**Figure 21 – Use-Case Diagram of Borrow Component**

**Brief Description:**
The user can use this use case only if s/he logins into system. S/he should use barcode reader to determine which books s/he wants to borrow.

**Step-by-step Description:**

1. The user clicks into Borrow with Barcode button and the system displays material info. page of the system

2. The user checks the material is suitable for borrowing or not

3. If the user has no penalty the user clicks the borrow button.
4. If the database updated successfully, you have successfully borrowed materials will be written on the screen. Otherwise, reason why the user cannot borrow material will be seen

### 4.5.2.2 Use Case: View Situations of Carells

**Diagram:**

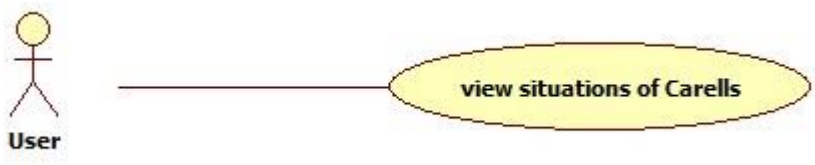


**Figure 22 – Use-Case Diagram of Carell Component**

**Brief Description:**

The user can use this use case whether s/he logins into system.

**Step-by-step Description:**

1. The user clicks into menu button and the system displays menu page of the system

2. The user selects the Carell in the Menu.

3. The system shows which carells are suitable for studying

## 4.6 Composition Viewpoints

Following component diagram shows the structural relationships between the components of our application. The diagram presents an early understanding of the overall system that is being built.

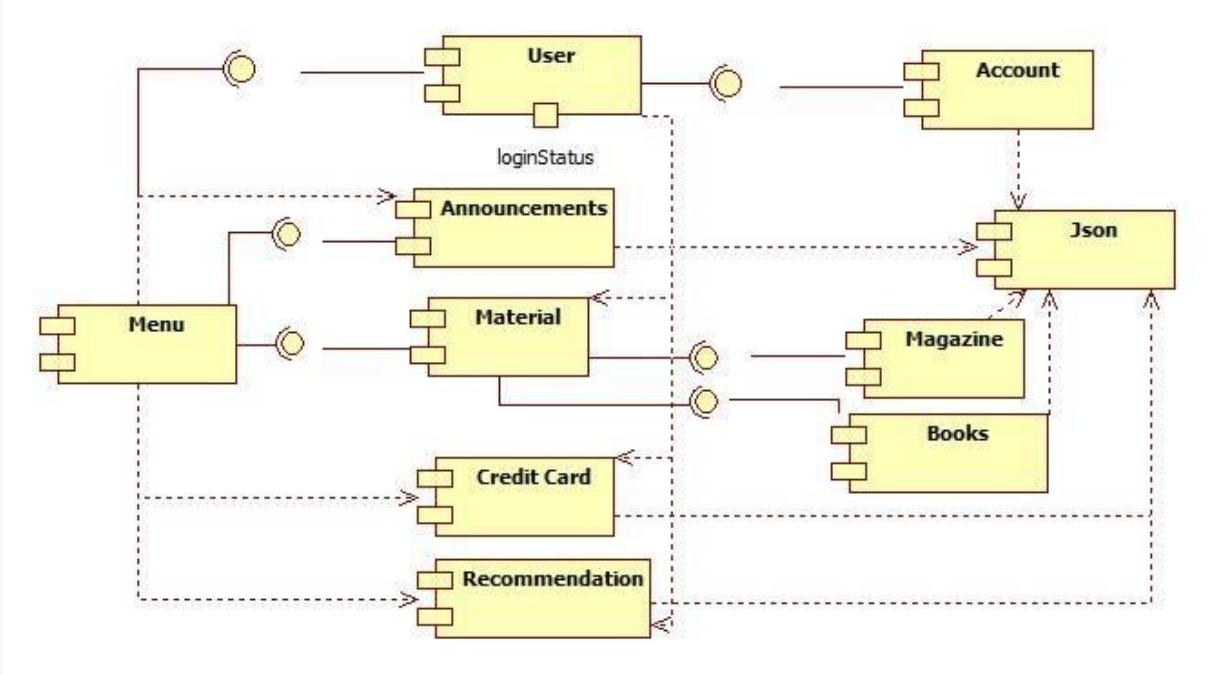


**Figure 23 – General Component Diagram**

### 4.6.1.1 Design Concerns

This viewpoint states information related to planning, monitoring and controlling the project. To give early view of the logical software components we provided a component diagram.

### 4.6.1.2 Design Elements

**Menu Component**

Design entity: type is class

Design relationships: this component needs Account, Announcements and Material components. They are connected with each other by realization relationship. In addition to these components, it also has dependency relationships with Credit Card and Recommendation components.

**User Component**

Design entity: type is class

Design relationships: this component needs Account component with a realization relationship. User component also has dependency relationship with Material, Credit Card and Recommendation components.

**Announcements Component**

Design entity: type is class

Design relationship: it has dependency relationship with Json component.

**Material Component**

Design entity: type is class

Design relationship: this component needs Magazine and Book components. They are connected with each other by realization relationship.

**Credit Card Component**

Design entity: type is class

Design relationships:  it has dependency relationship with Json component.

**Recommendation Component**

Design entity: type is class

Design relationship:  it has dependency relationship with Json component.

**Magazine Component**

Design entity: type is class

Design relationship:  it has dependency relationship with Json component.

**Book Component**

Design entity: type is class

Design relationship:  it has dependency relationship with Json component.

**Json Component**

Design entity: type is library

Design relationship: Account, Announcements, Magazine, Book and Recommendation components depends to this component.

**5. Requirements Matrix**

UC stands for use case.

| Use Cases From SRS | Login Component | Search Component | Suggest Component | View Announcement Component | Extend Materials Component | Hold Component | Pay The Debt Component | Logout Component |
|---|---|---|---|---|---|---|---|---|
| UC1 | X | | | | | | | |
| UC2 | | | | | | | | X |
| UC3 | | X | | | | | | |
| UC4 | | | X | | | | | |
| UC5 | | | | X | | | | |
| UC6 | X | | | | X | | | |
| UC7 | | X | | | | | | |
| UC8 | | | | X | | | | |
| UC9 | | X | | | | | | |
| UC10 | X | | | | | X | | |
| UC11 | X | | | | | | X | |
| UC12 | X | | | | | | X | |
| UC13 | | X | | | | | | |
| UC14 | | | X | | | | | |