COLLEGE OF INFORMATION TECHNOLOGY
UNIVERSITI TENAGA NASIONAL

UNITEN STUDENT TIME TABLE ON ANDROID PLATFORM WITH ASSIGNMENTS SCHEDULER

ALIBEK SABRALIYEV

2013

UNITEN Student Time Table on Android Platform with Assignments Scheduler

by

ALIBEK SABRALIYEV

Project Supervisor: Dr. Salman Yussof

A REPORT SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE BACHELOR OF
COMPUTER SCIENCE,
COLLEGE OF INFORMATION TECHNOLOGY
UNIVERSITI TENAGA NASIONAL

ABSTRACT

An online system is successfully used by Universiti Tenaga Nasional students and staff. However, time by time, any existing information system is needed to be maintained in order to satisfy requirements of its rapidly developing environment. This research is focused on developing and introducing an alternative way of keeping student timetable. Since the mobile technologies are occupying our everyday routine and become a part of people's lives, students will be able to keep timetable on their Android phones and another Android based mobile devices. Along with convenience in use, the application to be developed is going to adopt additional features like maintaining class assignments, importing, and export or import timetable.

TABLE OF CONTENTS

	Page
DECLARATION	ii
DEDICATION	iii
ABSTRACT	iv
TABLE OF CONTENTS	v
LIST OF TABLES	viii
LIST OF FIGURES	ix
CHAPTER 1 INTRODUCTION	
1.1 Project Background	1
1.2 Problem Statement	2
1.3 Objectives	2
1.4 Scope	2
1.5 Requirements	3
CHAPTER 2 LITERATURE REVIEW	
2.1 Introduction to Literature Review	4
2.2 Overview of the Mobile Application Development	4
2.2.1 Developing J2ME Applications	7
2.2.1.1 Introduction to J2ME	7
2.2.1.2 General J2ME Architecture	8
2.2.1.3 J2ME Development Tools	9
2.2.1.4 Advantages and Disadvantages of J2ME	9
2.2.2 Developing IOS Application	10
2.2.2.1 Introduction to IOS	10

2.2.2.2 IOS Architecture 10
2.2.2.3 IOS Development Tools 11
2.2.2.4 Features and Limitations of Developing on IOS 12
2.2.3 Developing Android Platform 13
2.2.3.1 Introduction to Android 13
2.2.3.2 Android Platform Architecture 13
2.2.3.3 Android Development Tools 16
2.2.3.4 Android Platform Features 17
2.3 General Overview of Uniten Current System 18
2.4 Review of Existing Applications 19
2.4.1 "A+ Timetable" Application 19
2.4.2 Timetable Application of Uniwersitet w Bialymstoku (UWB) 21
2.4.3 "Homework and Timetable" Application for Students 22
2.4.4 Comparison of Reviewed Applications 24
2.5 Development Tools and Programming Languages 25
2.5.1 Application Software 25
2.5.2 Software Development Languages 26
2.5.2.1 Java 26
2.5.2.2 SQL 26
2.5.2.3 PHP 27
2.5.2.4 HTML 27

2.6 Reviews on Development Methodologies 27
2.6.1 Waterfall Software Development Methodology 28
2.6.1.1 Waterfall Strengths 29
2.6.1.2 Waterfall Weaknesses 29
2.6.2 Iterative Software Development Methodology 30
2.6.2.1 Iterative Strengths 30
2.6.2.2 Iterative Weaknesses 31

CHAPTER 3 ANALYSIS

3.1 Chosen Software Development Methodology 32
3.2 The Detailed Structure of Waterfall Model Software Development Methodology and Phases 33
3.3 Requirements Engineering 35
3.3.1 Reviewing Existing Applications 36
3.3.2 Interview 36
3.4 System Design 39
3.5 Software Development Tools and Programming Language 39

CHAPTER 4 DESIGN

4.1 An Overview of the Application Design 41
4.1.1 Application Design Specifications 41
4.2 Flowchart of the Proposed Application 43
4.3 System Design 46
4.3.1 Data Design 46
4.3.2 Database Design 49
4.3.3 Interface Design 51

CHAPTER 5 IMPLEMENTATION

5.1 Physical Design 59
5.2 Technical Design 59
5.3 Coding 72
5.3.1 Database Handler 72
5.3.2 Timetable representation 72
5.3.3 Implementation of inter-activity communication 73
5.3.4 Android native widgets 76
5.4 Timetable export and import 81

CHAPTER 6 TESTING AND VERIFICATION

6.1 Introduction to application testing 84
6.2 Unit testing 84
6.2.1 Create timetable function 85
6.2.2 Edit time table function 87
6.2.3 Add new assignment function 88
6.2.4 Import timetable function 89
6.3 Usability testing 91
6.3.1 Heuristic evaluation 91
6.4 Integration testing 93
6.4.1 Creating, importing and updating timetable 93
6.4.2 Adding assignment and assignment notification 94
6.4.3 Putting altogether 94

CHAPTER 7 CONCLUSION

7.1 Project Achievements 95
7.2 Problems Encountered 96
7.3 Future Enhancements 97

REFERENCES 98

APPENDICES

APPENDIX A: Gantt Chart 100

APPENDIX B: List of Standard Abbreviations 101

LIST OF TABLES

Table No.	PAGE
2.1 Comparison of Reviewed Applications	24
3.1 Interview Results	37
4.1 Timetable Information	49
4.2 Assignment Information	50
6.1 Create timetable test case	85
6.2 Edit timetable test case	87
6.3 Add new assignment test case	88
6.4 Import timetable test case	90
6.5 Heuristic evaluation	91

LIST OF FIGURES

Figure No.	Page
2.1 Small Collection of Smartphones	5
2.2 The number of downloads from Android Market	6
2.3 J2ME Architecture	8
2.4 IOS application layer architecture	11
2.5 Layers of IOS	11
2.6 Android architecture	14
2.7 "A+ timetable" application	20
2.8 Timetable application of UWB	21
2.9 "Homework and Timetable" application	23
2.10 Waterfall sequences	28
2.11 Iterative Methodologies	30
3.1 Waterfall Model Structure	33
4.1 Application Flowchart	44
4.2 Flowchart Create Timetable	45
4.3 Context Diagram	46
4.4 Level-0 DFD	47
4.5 ERD	48
4.6 Main Menu UI	51
4.7 View timetable interface	52
4.8 View assignments interface	53

4.9 Add assignment interface	54
4.10 Timetable creation interface step 1	55
4.11 Timetable creation interface step 2	56
4.12 Timetable creation interface step 3	57
4.13 Timetable creation interface step 4	58
5.1 Main menu	60
5.2 View timetable	61
5.3 Edit timetable	62
5.4 User notification	64
5.5 Create timetable step 1	65
5.6 Create timetable step 2	66
5.7 Create timetable step 3	67
5.8 Entering venue information	67
5.9 Confirmation of entered data	68
5.10 View assignments	69
5.11 Add assignments	70
5.12 Options	71
5.13 Insert method in Database Handler class	73
5.14 View Timetable XML	74
5.15 View Timetable java class	75
5.16 Intents in Android	76
5.17 Toasts in Android	76
5.18 Toast in action	77
5.19 Alert-box implementation	77
5.20 Alert-box in action	78
5.21 Notification Manager	78
5.22 Auto-silent mode implementation	79

3.2.5 Summary view 80
3.2.6 Data packet view 81
3.2.7 It speed simulation 82
3.2.8 Improved simulation 83

CHAPTER 1

INTRODUCTION

1.1 Project Background

Universiti Tenaga Nasional uses centralized system, which almost fully provides services in organizing, controlling and managing academic study and all processes related to it. Existing system is universal, because of its reliability and simplicity in use as well as a great variety of functionality exists. Many processes from student registration to finance transactions are handled and processed by this system. As time goes further, requirements of rapidly developing environment are getting higher. There is a need today, for students, to have an alternative way of keeping their timetable. Since, any system time by time should be maintained, new functionalities and features are introduced [1]. The research to be done is focused on analyzing the current system and introducing the new capability by developing an Android based application in order to provide students with more convenient way of keeping class schedule and managing their study.

1.2 Problem Statement

One of the most important and frequently used function of the current system among students is time table. Since, it is difficult to memorize time table, classes, venue and time of the class; students have to always refer to the schedule. Actually, there are many ways to keep or view the time table: login to the system, print time table, make screenshot and etc. All of those methods are workable, but have some drawbacks. Logging in requires network connection and PC/Laptop, printing time table waste paper, might be forgotten or lost, screenshots, if there is a change in the schedule, forces the student to make it again.

1.3 Objectives

1) To develop an Android application which can be used by students to create and keep class timetable, as well as main the schedule of class assignments.
2) To develop a Web-based timetable application similar to the one used by Uniten.

1.4 Scope

1) The application is developed based on classes at Universiti Tenaga Nasional.
2) The application is compatible with Android based devices.
3) The application will contain the following features:

    - Timetable can be created within the Android application itself.
    - Timetable can also be exported and imported through Internet/Bluetooth from a Web-based application.

The list of class assignments to be done can be created and maintained within the Android application. Some of the other futures are going to be discovered during the further research.

1.5 Requirements

The application to be developed needs following requirements:

- Eclipse IDE.
- Android simulator.
- Mobile phone with android operating system.
- Wamp server.
- Notepad++.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction to Literature Review

This chapter discusses related topics required for developing this mobile application. The study covers a general overview on mobile applications and mobile application development process, by reviewing the most popular development platforms. Also the discussion is done on the overall look of Universiti Tenanga Nasional (Uniten) current online system, review and comparison of existing mobile applications on Android platform and development tools and software developing language to be used for the development process itself.

2.2 Overview of the Mobile Application Development

Nowadays, there are around five billion people that use mobile phones. Mobile phones are advancing other technologies in terms of rapid speed of development that industry is experiencing. For comparison, there are only 1.7 billion personal computers being used

around the world [2]. According to Meier [3] now, it is an exciting time for mobile application developers, because mobile devices have never been more popular than now. Many different types of smartphones, tablets and PDAs are getting more preferable and becoming a regular choice for consumers. There are many features and applications available among mobile phones and devices such as, email sending, web, Global Positioning System (GPS), games, multimedia, entertainment, study and business. A number of companies like Apple, Nokia, Samsung, HTC, Google, RIM and Microsoft are leading in smart phones industry. **Figure 2.1** illustrates some types of smartphones.

Figure 2.1 Small Collection of Smartphones [4]

So, what makes popular the development of mobile applications? As mentioned in this chapter before, versatile and stylish mobile devices with features like GPS, touch screens,

accelerometers, powerful processors and huge memory provide an amazing platform upon which programmers and developers are becoming able to create innovative mobile [3]. If refer to online resources, there are around five hundred thousands of applications in each Apple Store and Android Market. **Figure 2.2** shows the number of downloading mobile applications by users from Android Market.

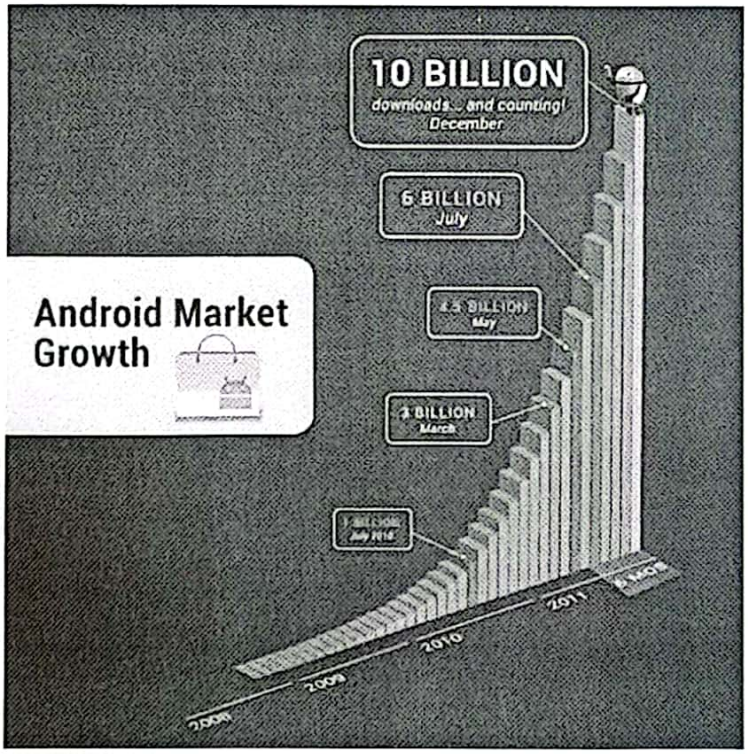


Figure 2.2 The number of downloads from Android Market [5]

From this huge numbers of statistics it seems that up until today the development in mobile applications field has significantly grown. Roughly say, it is obvious that per each developed application on Android, there are up to two thousands downloads. Again, it

proves that developing mobile applications is very popular from perspective of programmers as well as from perspective of users.

This research further discusses the three major platforms that are widely used in developing most mobile applications.

2.2.1 Developing J2ME Applications

This section will briefly explain about the main aspects in developing J2ME applications and about J2ME platform itself.

2.2.1.1 Introduction to J2ME

J2ME stands for Java 2 Micro Edition. Sun Microsystems defines J2ME as "a highly optimized Java run-time environment targeting a wide range of consumer products, including pagers, cellular phones, screen-phones, digital set-top boxes and car navigation systems." In another words, it is a version of Java designed for devices with limited memory and processor resources, which makes it as an ideal environment for developing software on mobile devices [6].

2.2.1.2 General J2ME Architecture

The general architecture of J2ME can be considered modules divided into two main building blocks: configuration and profile as shown in **Figure 2.3**.

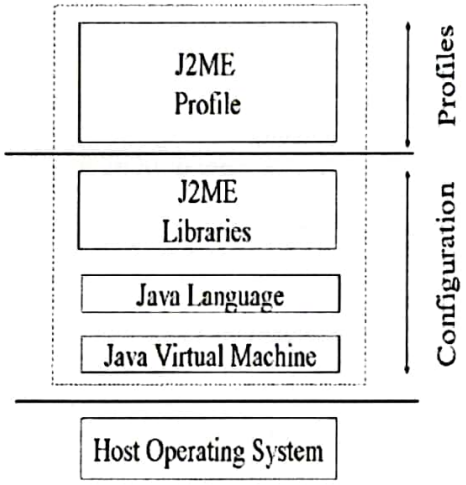


Figure 2.3 J2ME Architecture [6]

Configuration: J2ME configuration defines minimum requirements of the platform needed for group of devices which have similar processing and memory capabilities. A configuration is based on Java Virtual Machine (JVM), java language and minimum class libraries in order to support that group of devices [6].

Profile: A profile extends configuration and responsible for specific needs of the certain device family. It acts as a mean which guarantee interoperability between devices and comes with a variety of Java class libraries [6].

2.2.1.3 J2ME Development Tools

J2ME applications as normal Java applications are able to be developed on any IDE like Eclipse or Netbeans. The only requirement is being aware of libraries to be used. Also, there are several types of IDEs for special purposes such as developing wireless networking applications [6]:

- Borland JBuilder 7 Enterprise with MobileSet 3.
- Sun Microsystems' Sun ONE (Open Network Environment) Studio 4 Mobile Edition.
- Metrowerks CodeWarrior Wireless Studio 7.
- S5 Systems' jVise (based on IBM Eclipse technology).

2.2.1.4 Advantages and Disadvantages of J2ME

Wide variety of cellphones like Nokia, earlier versions of Samsung support J2ME applications. However, as any technology available today, J2ME has its own advantages and disadvantages. The main disadvantage is the fact that not many mobile devices nowadays support J2ME applications, in other words this platform is not so popular among the users [6].

2.2.2 Developing IOS Applications

This section will briefly explain about the main aspects in developing IOS applications and about the IOS platform itself.

2.2.2.1 Introduction to IOS

IOS previously stands for IPhone Operating System, which is developed and distributed by Apple Inc. As IPhones, IPads and IPods are getting more popular and popular; the interest in developing applications among programmers also grows [7].

2.2.2.2 IOS Architecture

The IOS architecture is similar to Macintosh OS architecture. The top of IOS layer organized as in the **Figure 2.4**, which shows that IOS acts as an intermediary layer, so that it can be suitable for different types of hardware.

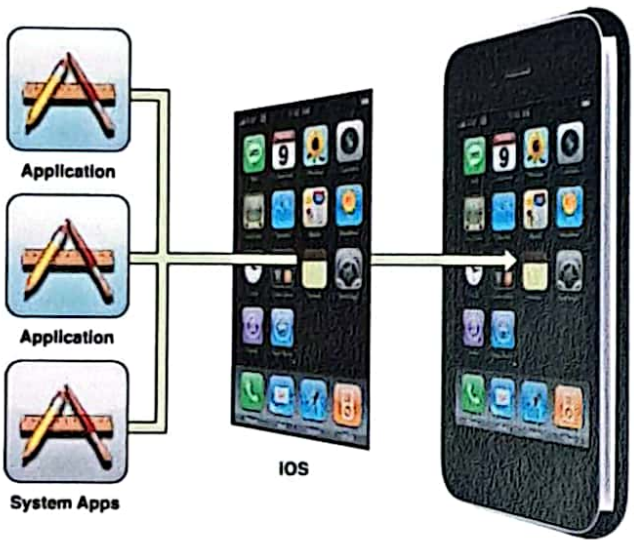


Figure 2.4 IOS application layer architecture [7]

But, the general implementation of IOS technology can be viewed as layers as in **Figure 2.5**. At the lower levels the fundamental services run and at the higher levels more sophisticated services and technologies are implemented.

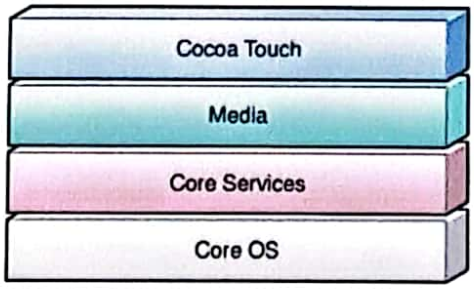


Figure 2.5 Layers of IOS [7]

2.2.2.3 IOS Development Tools

The complete toolset for developing IOS applications includes [7]:

- XCODE IDE.
- Performance analysis tools.
- IOS simulator.
- Mac OS.
- IOS SDKs.

2.2.2.4 Features and Limitations of developing on IOS

According to Goldstein [8] there are features and limitations in developing applications on IOS. The main features are:

- Access the Internet.
- Know the location of the user.
- Track orientation and motion.
- Track the action of the user's finger on the screen.
- Play video and audio.
- Access the user's contacts.
- Access the user's pictures and camera.

Along with powerful features, the IOS also has some limitations such as [7]:

- Small screen of IPhones.
- Inconvenience for users with fat fingers.
- Limited processor power, memory and battery life.

2.2.3 Developing on Android Platform

This section will briefly explain about the main aspects in developing Android applications and about the Android platform itself.

2.2.3.1 Introduction to Android

Android is the world popular platform based on Linux OS developed by Open Handset Alliance, led by Google. Many mobile devices like smartphones, tabs and notes under world famous brands as Samsung, HTC, Motorola and LG run on this platform [8].

2.2.3.2 Android platform architecture

The architecture of Android platforms is based on smaller version of Linux OS and its kernel. This section describes the architecture of Android platform according to Meier [3] and Android-Sharks online resources web-site. **Figure 2.6** illustrates the general overview of the Android architecture.

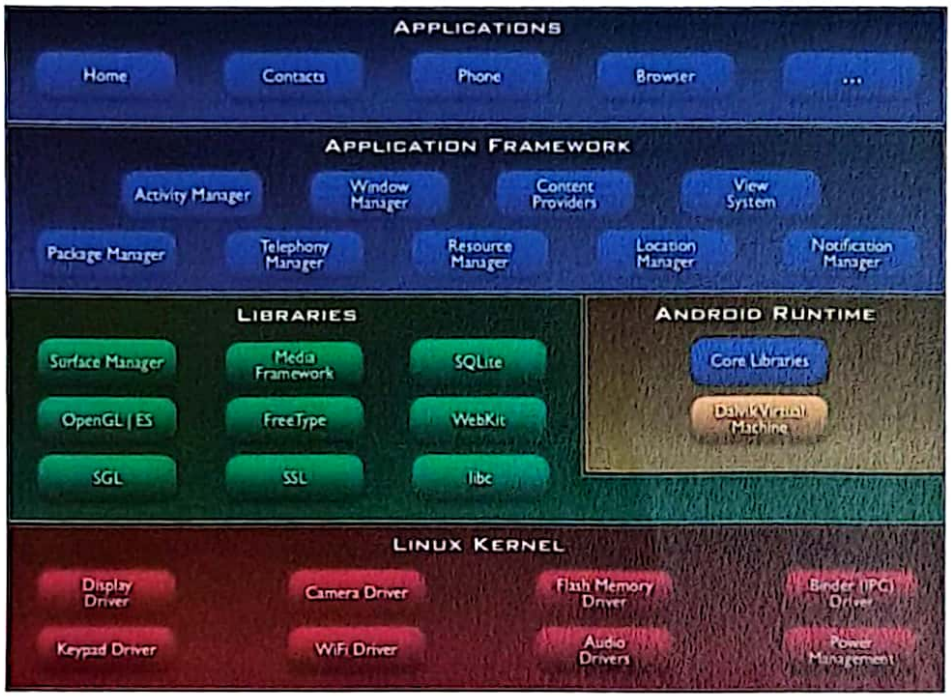


Figure 2.6 Android architecture [9]

As the figure shows, the basic fundamental part of the architecture is a Linux kernel, where it ensures the functioning of the system and is responsible for security, memory management, power system and processes, and provides a network stack and model of drivers. Kernel also acts as an abstraction layer between hardware and software stack. "Above" kernel as software middleware is a set of libraries, designed to provide essential basic functionality for applications. That is, this level is responsible for providing the implemented algorithms for higher-level, support for file formats, the implementation of encoding and decoding of information (in the example is a multimedia codecs), graphics rendering, and more. Libraries are implemented in C / C++ and compiled for a specific

hardware device, with which they are supplied by the manufacturer in the form of a preset [9].

Given below are several examples of Libraries used in Android are listed [9]:

- **Surface Manager** - is a composite window manager like Compiz (Linux), but more simplified. This allows the system to create interesting seamless effects, transparent windows, and smooth transitions.
- **Media Framework** - library, implemented based on PacketVideoOpenCORE. With their help, the system can record and playback audio and video content, as well as the output of static images. It supports many popular formats, including MPEG4, H.264, MP3, AAC, AMR, JPG, and PNG.
- **SQLite** - a lightweight and high performance relational database that is used in Android as the main engine for working with databases used by applications to store information.

On the same level is Android Runtime. Its key components are the core set of libraries and virtual machine Dalvik. Libraries provide most of the low-level functionality that is available to libraries of the Java [9].

Level is located above the Application Framework, sometimes called frame level applications. Through the application framework developers get access to the API,

provided by the system components that lie below the level. In addition, due to the architecture of the framework, already realized features of other applications are able to be accessed, unless access is allowed [9].

At the top of the Android software stack is the application layer (Applications). This includes a set of basic applications, which are pre-installed on OS Android. For example, it includes a browser, email client program for sending SMS, maps, calendar, contact manager, and many others. List of integrated applications can vary depending on model and version of Android. And apart from this basic set of applications, all other applications including those that are installed by the user are able to be taken under level of concern [9].

2.2.3.3 Android Development Tools

The main tools used in developing Android applications are:

- Android Software Development Kit (SDK).
- Indegrated Development Environment (Eclipse, NetBeans).
- Android Development Tools (ADT, plugin).
- Dalvik Virtual Machine.
- Android Simulator.

2.2.3.4 Android platform features

According to Meier [3], Android Platform has these noteworthy features that are listed below:

- No licensing, distribution, or development fees.
- Wi-Fi hardware access.
- GSM, EDGE, and 3G networks for telephony or data transfer, allowing you to make or receive calls or SMS messages, or to send and retrieve data across mobile networks.
- Comprehensive APIs for location-based services such as GPS.
- Full multimedia hardware control including playback and recording using the camera and microphone.
- APIs for accelerometer and compass hardware.
- IPC message passing.
- Shared data stores.
- An integrated open source WebKit-based browser.
- Full support for applications that integrate Map controls as part of their user interface.
- Peer-to-peer (P2P) support using Google Talk.
- Mobile-optimized hardware-accelerated graphics including a path-based 2D graphics library and support for 3D graphics using OpenGL ES.
- Media libraries for playing and recording a variety of audio/video or image formats.

- An application framework that encourages reuse of application components and the replacement of native applications.

2.3 General Overview of UNITEN Current System

As was mentioned before Universiti Tenaga Nasional uses an online system in order to organize academic process and all other things related to it. An interview was conducted with Manager, Database Administrator of IT and Multimedia services of Uniten Mr. Mohammad Shukeri bin Yusuff. Basically, all questions were focused on technical part of the system in general and the main attention was paid on the organization of the timetable.

Uniten uses this system in almost all its working areas. These fields include education, management, finance, technicians and others. All processes are interrelated and refer to the general database, which have been developed using Microsoft SQL Server. The system has a multileveled an authorized access and each level has it is own priorities and restrictions. The authorization process is processed by using Windows Active Directory, means that the database does not store any password. A specific Graphical User Interface has been developed on Visual Basic 6, which is also responsible to process and represent the data from the database.

The discussion is going to be continued in more detail in Chapter 3 in order to identify necessary requirements.

2.4 Review of Existing Applications

In this section several types of existing applications with similar functions are going to be discussed.

2.4.1 "A+ Timetable" application

According to online sources "A+ timetable" is an application developed on Android platform which helps to organize daily schedule in the school. **Figure 2.7** illustrates the screenshot of the main interface of the "A+ timetable" application [10].

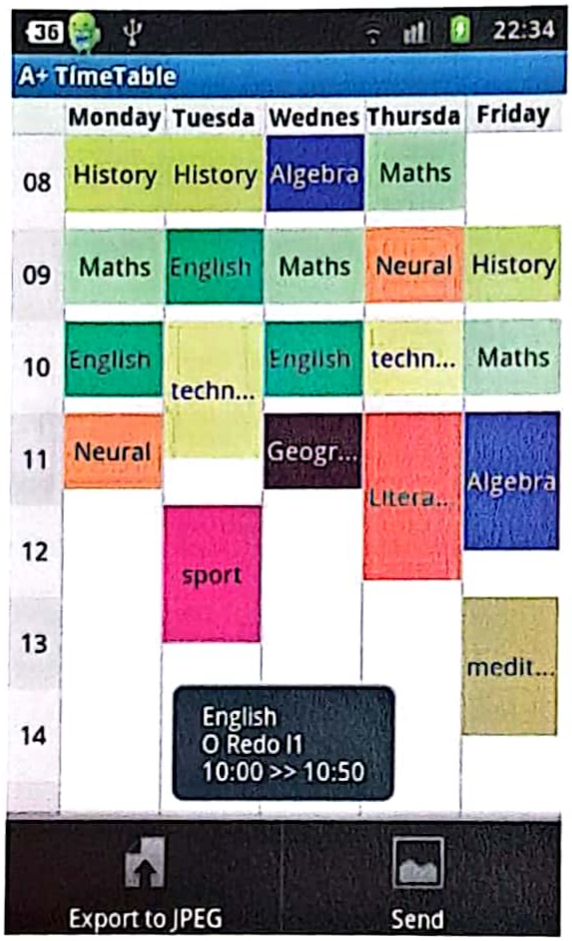


Figure 2.7 "A+ timetable" application [10].

The "A+ timetable" has the following features:

- Week overview.
- Widget with current/next.
- Odd/Even week option.
- Tab days view.
- Text to speech.

- Export timetable as image.
- Auto silence/vibrate phone during lessons.
- Share timetable with friends (Bluetooth, email).

2.4.2 Timetable Application of Uniwersitet w Bialymstoku (UWB)

This application has been developed for Android platform for servicing students of Poland University UWB in order to provide with opportunity to view the timetable and details of the subject. **Figure 2.8** demonstrates the screenshot of the main view of the application [11].

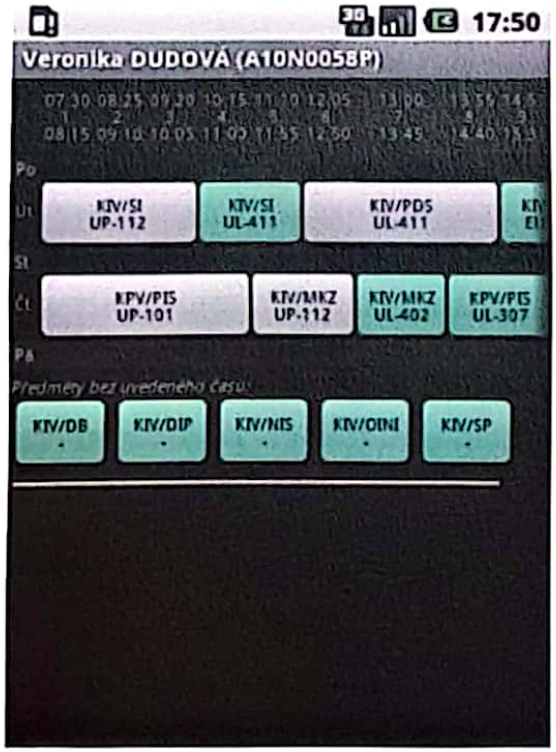


Figure 2.8 Timetable application of UWB [11].

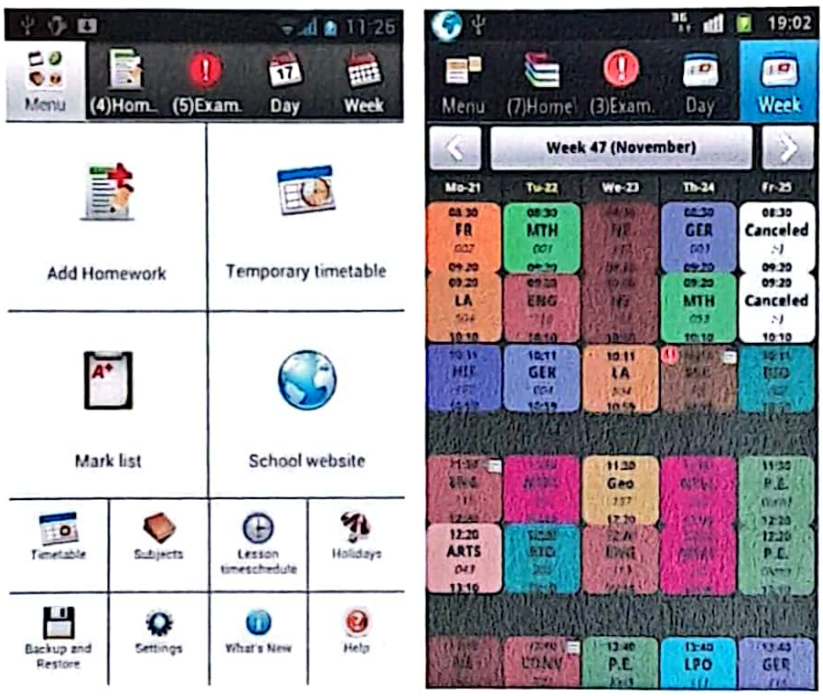


Figure 2.9 "Homework and Timetable" app. [12]

This application has several features listed below:

- Create timetable with homework assignments and exam.
- Maintain timetable changes.
- Rotating screen.
- Unfinished assignments notification.
- All subjects, assignments and exams follow real date.
- Auto silent mode.

Application allows to:

- View the schedule according to personal number.
- Enter more personal numbers and switch between them.
- View subject details by clicking on them.
- View lecturers' details.
- Widget.
- Gesture for switching.

2.4.3. "Homework and Timetable" Application for Students

The "Homework and Timetable" is another application available for Android phones which is able to keep track of student's current class schedule and homework assignments. The **Figure 2.9** shows the screenshot of this application [12].

2.4.4 Comparison of Reviewed Applications

Three existing similar applications have been reviewed in the sections above. **Table 2.1** compares the features of the existing application with the proposed one.

Table 2.1 Comparison of Reviewed Applications

Features of the applications	A+ Timetable	Timetable of UWB	Homework and Timetable	Proposed application
Timetable creation	Yes	No	Yes	Yes
Timetable editing	No	Yes	Yes	Yes
Importing timetable	No	Yes	No	Yes
Export/backup timetable	Yes	No	Yes	Yes
Assignments keeping	No	No	Yes	Yes
Auto Silent Mode	Yes	No	Yes	Yes
Screen Rotation	No	No	Yes	Yes

2.5 Development Tools and Programming Languages

2.5.1 Application Software

Eclipse is an open source community whose projects building tools and frameworks for creating general purpose application. The most popular usage of Eclipse is as a Java development environment (eclipse.org).

NetBeans IDE is a free open-source program for developers in order to create professional desktop, enterprise, web and mobile applications (netbeans.org)

Android Development tool is a software development kit used as a plugin within the Eclipse IDE. It allows programmers to create and debug applications easier and faster.

Notepad++ is a free source code editor. It is somewhat similar with Notepad and WordPad. The main differences from WordPad are that it allows users to manage different documents in tabulated form. In addition, it also provides line-numbering which is important for debugging purposes. The interface mainly consists of whitespace for codes which means that user has to compile the codes and only be able to see the interface. [13]

2.5.2.3 PHP

The term PHP stands for Hypertext Preprocessor is a widely-used open source general-purpose scripting language that is especially suited for web development and can be embedded into HTML. What distinguishes PHP from something like client-side JavaScript is that the code is executed on the server, generating HTML which is then sent to the client. The client would receive the results of running that script, but would not know what the underlying code was. Users can even configure a web server to process all HTML files with PHP [16].

2.5.2.4 HTML

HTML stands for Hypertext Markup Language, developed by scientist Tim Berners-Lee in 1990, HTML is the "hidden" code that helps us communicate with others on the World Wide Web (WWW).

When writing HTML, users add "tags" to the text in order to create the structure. These tags tell the browser how to display the text or graphics in the document [17].

2.6 Reviews on Development Methodologies

This section provides a brief review on the most common two types of software development methodologies such as Waterfall and Iterative Methodologies.

2.6.1 Waterfall Software Development Methodology

The waterfall emphasizes a structured relation between defined phases. And each phase, which consists of certain set of activities, must be completed before proceeding to the next phase [18]. Primarily, waterfall methodology divides the project into concrete phases with predefined outcomes for each phase. Mostly each company or project team name these phases differently, according to their project. But generally, waterfall methodology can be viewed as shown in the **Figure 2.10**.

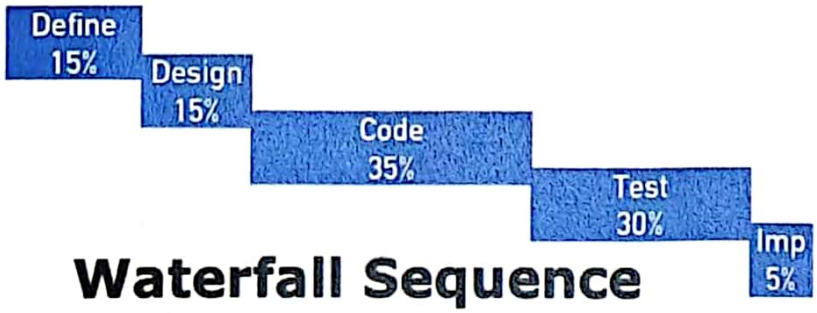


Figure 2.10 Waterfall sequences [18]

The first phase is about what the system will do, in other words its system requirements. Second, how the project is going to be designed. Third, is about actual coding. The fourth phase is focused on testing the developed system or application. And finally, the fifth phase is totally about tasks such as go-live, documentation and training.

2.6.1.1 Waterfall Strengths

Most of the benefits from using a waterfall methodology are directly related to its underlying principles of structure. These strengths include [18]:

- Ease in analyzing potential changes
- Ability to coordinate larger teams, even if geographically distributed
- Can enable precise dollar budget
- Less total time required from Subject Matter Experts

2.6.1.2 Waterfall Weaknesses

Along with its advantages, highly structured approach also leads to certain disadvantages such as the following [18]:

- Lack of flexibility.
- Hard to predict all needs in advance.
- Intangible knowledge lost between hands-offs.
- Lack of team cohesion.
- Design flaws not discovered until the Testing phase.

2.6.2 Iterative Software Development Methodology

The Iterative methodologies are focused on highly concentrated teams with minimum structure and constant feedback form Subject Matter Experts [18]. **Figure 2.11** illustrates the sequences involved during the iterative software development methodology.

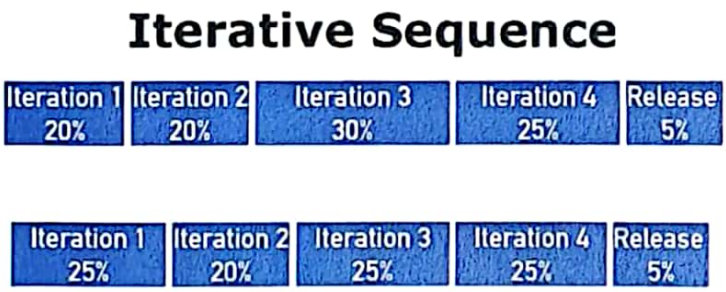


Figure 2.11 Iterative Methodologies [18]

The iterative methodology is focused on creating a series of working prototypes for evaluation by the SMEs until the objectives are accomplished and the system is ready for final release. In other words, during the iterative methodology the essential team keeps creating prototypes until all requirements are satisfied and goals are met.

2.6.2.1 Iterative Strengths

Many of the strengths of the iterative approach are listed below [18]:

- Rapid feedback from actual users.
- Flexibility to address evolving requirements.
- Design flaws discovered quickly.
- Easy to roll-out new functionality in stages.

- Higher motivation and great productivity.
- Very little knowledge loss between phases.

2.6.2.2 Iterative Weaknesses

The drawbacks to using an iterative approach are worth considering and should be weighed carefully when deciding on a methodology for a new project. Some of the more serious weaknesses include [18]:

- Difficulty in coordinating larger teams.
- Can result in a never-ending project if not managed properly.
- Tendency to not document thoroughly.
- Predicting the precise features to be accomplished in a fixed time/budget.

CHAPTER 3

ANALYSIS

3.1 Chosen Software Development Methodology

A software development methodology is a framework used to organize, plan, structure and control the process of developing an application or information system [19]. Software development process may consist of research, new development, prototyping, modification, reuse, re-engineering, maintenance or any other activities that result in software product. Two the most common methodologies waterfall model and iterative approach were reviewed in Chapter 2. The waterfall software development methodology was chosen as the most suitable methodology in order to develop the application proposed and satisfy all requirements of the project.

3.2 The Detailed Structure of Waterfall Model Software Development Methodology and Phases

As was mentioned before the structure of waterfall model consists of discrete phases. The name, outline and outcome of each stage depend on the project. **Figure 3.1** illustrates the whole structure of the complete waterfall model being used throughout this project.

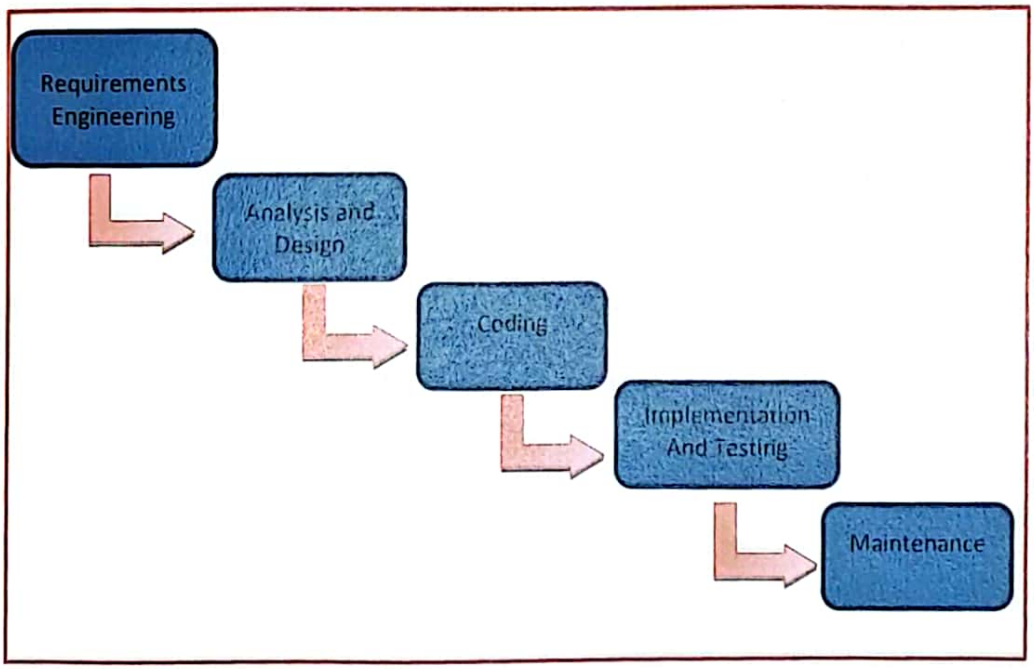


Figure 3.1 Waterfall Model Structure

As it is shown above, the application to be developed is going through these stages. Each stage is defined by the brief description further in this section.

Requirements Engineering – is a first stage of a whole lifecycle of project development where the requirements of the proposed system are collected. The methods used for collecting data are interview and observation. The output of this stage is to generate Requirements Specification Document.

Analysis and Design – is second stage where and architectural design is created and main components of application, together with interfaces and their interaction are identified. Along with producing soft architecture of the application, the hardware such as mobile device platform, operating system and other peripheral devices are chosen. The output of this phase is to generate System Architecture document (Flowchart, Sequence, Entity Relationship, Data Flow and etc. diagrams).

Coding – is the stage where the actual coding takes place. The application is going to be coded according to software design document. The programming language and environment, coding technique and code documentation are identified. The output of this stage is Executable Software Application.

Implementation and Testing – is a stage where developed application will be installed on several mobile devices and properly tested. Throughout this stage the application is going to be tested from different approaches and prepared for the final release. The application

to be released must satisfy its initial requirements. The output of this phase is Integrated Software Product.

Maintenance – is a final stage where successfully installed application is going to be periodically checked for the identifying updates if necessary. The changes will take place according to periodic feedback collected from users or according to other environmental changes in application requirements such us, changing the timetable format of UNITEN, format of subject code or subject name as well. The output of this phase is report on the needs for updating, followed by updated version the application.

However, the scope of the Project 1 is to cover Requirements Engineering and System Analysis and Design phases.

3.3 Requirements Engineering

In order to identify sufficient requirements to develop the proposed application an interview with specialist from IT and Multimedia Services department of Uniten was conducted and review on existing similar applications was done.

3.3.1 Reviewing Existing Applications

Reviewing existing system is one of the data collection tools that had been applied for requirement gathering. The reviews have been discussed in Chapter 2, section 4. From that it has been decided that the application to be developed will have the main features such us:

- *Timetable creation:* allows users to create timetable.
- *Timetable editing:* allows users to edit timetable.
- *Importing timetable:* allows users to import timetable from Uniten system.
- *Export/backup timetable:* allows users to export timetable as JPEG file.
- *Assignments keeping:* allows users to keep records about class assignments.
- *Auto Silent Mode:* automatically switch in silent/vibrate mode if it is class time.
- *Screen Rotation:* supports screen rotation to comfortable timetable view.

3.3.2 Interview

As was mentioned earlier in Chapter 2, an interview was conducted on July 26, 2012 at ITMS of Uniten with manager and database administrator Mr. Mohammad Shukeri Bin Yusuff. The questions asked, were focused on the structure and organization of the

timetable of Uniten online system. **Table 3.1** contains interview questions and answers correspondingly.

Table 3.1 Interview Results

Interview Questions	Answers
What can you say generally about Uniten System, from the perspective of the student?	In order to authenticate students the system uses Microsoft Active Directory. Database which is on Microsoft SQL Server doesn't store any username and password. After logging in to the directory, students can easily access their data online, including timetable and etc.
What are the main components of whole system, from the perspective of the developer?	As I said, we use active directory in authentication process, the Database Management System is built on Microsoft SQL Server and the Graphical Interface is organized by using Visual Basic 6.
How the data in the timetable is organized? From database point of view.	The table in the database consists of several fields, which are of course interrelated with others. And this data is accessed by using VB 6 and represented as students can see now.
What algorithm is used to represent the timetable?	We use numbers to mark days, i.e. 1 – Sunday, 2 – Monday and etc. and use starting time of the session and duration. This simple structure allows us to represent time table properly.

Is it possible somehow integrate or to embed my proposed application with Uniten system?	Yes, technologically it is possible. But, I'm not sure how exactly this process will be handled. Should your application connect to our system or we should embed somehow your application. This is research you should do.
Do you have any raw idea about how to organize transfer of the data from the system to mobile application? In what form?	Better to send textual data, since any mobile application faces lack of resource, especially memory. But again, what is exactly way to send data, you should think.

Overall, six question were asked during the interview. Based on the answers, following requirements are identified:

- The data entries in database of mobile application must correspond to the data in Uniten system timetable.
- The DBMS in mobile application must use SQL language.
- The application must be able to properly read data from the textual format and correspondingly transfer it into correct form of the timetable.
- The application must provide the easiest and smooth way with minimum overhead in order to import timetable from Uniten System.

3.4 System Design

During the requirements engineering process all requirements of the application are identified. All the analyzed requirements have to be structured and converted into organized and useful foundation in order to proceed with the application design. In the System Design stage, all the prepared and analyzed system description is going to be used to build logical and physical system specifications. The outcome of the this stage is a variety types of diagrams such as Flowchart, Data Flow Diagrams(DFD), Entity Relationship Diagrams(ERD), Sequence Diagrams and etc. which are going to be discussed in Chapter 4, System Design.

3.5 Software Development Tools and Programming Language

Eclipse IDE together with Android Developing Tools (ADT) plugin is chosen as the main tool in order to develop the proposed application. Based on the research that has been done in Chapter 2, Eclipse became the most suitable tool for developing an application on Android. ADT is a full functional plugin, which contains Android Simulator in order to immediately test the developing application. Also many tutorials are available in the Internet in order to learn and enhance the code being developed.

The Java programming language is chosen as the main language for the developing the application. But, still there are secondary languages which will be used in developing application such as HTML and PHP in order to create dummy page similar to Uniten's

one, XML in order to create the interface of the application and SQL in order to process data manipulations and operation.

The Database Management System to be used will be SQLite. It is quite popular and widely used DBMS among the mobile application developers.

CHAPTER 4

DESIGN

4.1 An Overview of the Application Design

The Application to developed is going to satisfy todays existing necessity among students in order to have an alternative, easy and convenient way of keeping and maintaining class timetable. In order to achieve this, the proposed application should be properly designed. This chapter covers three main components of the application design such as data, database and interface designs. Also, the flowcharts and some design specifications are provided.

4.1.1 Application Design Specifications

Also, there are some specifications needed to be noted during the database design and actual coding. The objective of designing the database is to make the structure of the data

as close to Uniten timetable as possible. In order to come up with correspondingly similar structure of the data and appropriate data fields inside the database the requirements from the interview are analyzed. The following points are essential to be considered during the application design and coding itself:

- The Data Flow Diagrams must be drawn according to Uniten timetable data structure.
- The data fields in Entity Relationship Diagram must correspond to Uniten timetable data fields.
- The programming language to be used in developing database must be SQL.
- The correspondence must result in easy and smooth transfer of timetable data from the system to the application.

The process of the importing timetable from Uniten system is a big deal needed to be considered from many perspectives like possibility to implement, necessity to change Uniten system, cost, security issues and etc. But, at this level, on theory, the importing timetable mechanism is going to be implemented in the following way:

1. The Import button will connect user to Uniten system;
2. The Uniten system will prompt user to enter his ID and Password;
3. The Export button will be available on the Uniten System and will save the timetable data in a certain format on the mobile phone;
4. Finally the application will load the data into database and display the complete timetable to the user.

4.2 Flowchart of the Proposed Application

Flowchart is a graphical or symbolic representation of the processes involved into the whole functioning of the application. **Figure 4.1** shows the flowchart of the application to be developed.

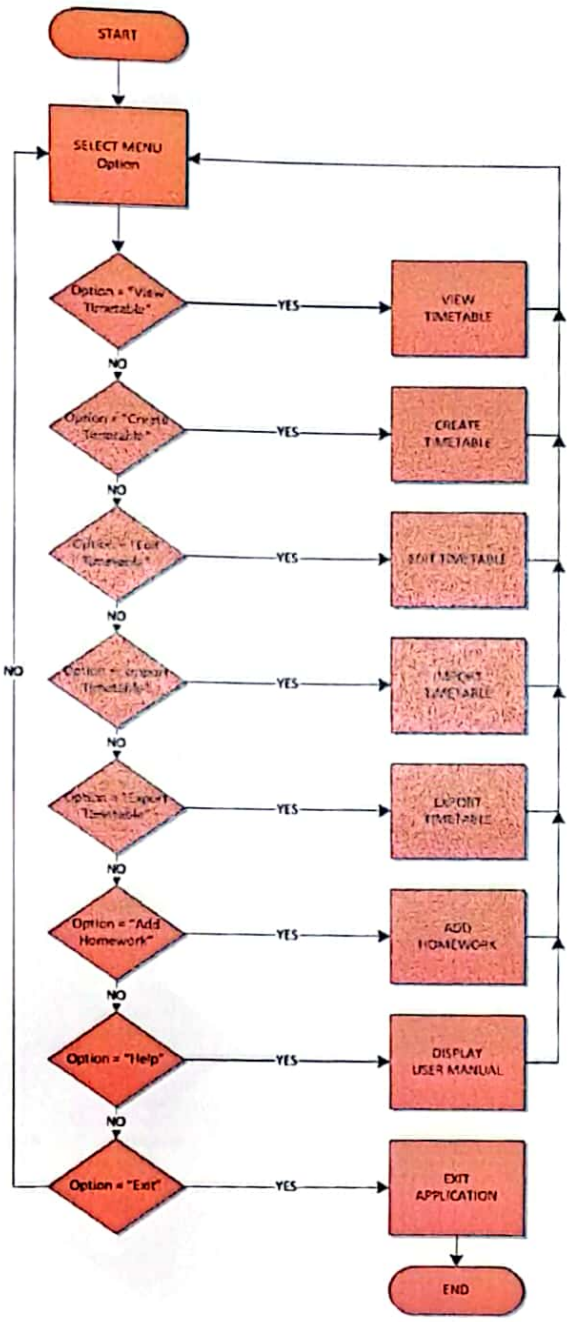


Figure 4.1 Application Flowchart

Once a user launches the application, the main menu with several options appears. As it is shown in the flowchart above eight options are available for the user to choose. Each option is followed by a corresponding set of actions. By now, only create timetable option is designed, **Figure 4.2** is the flowchart of create timetable option.

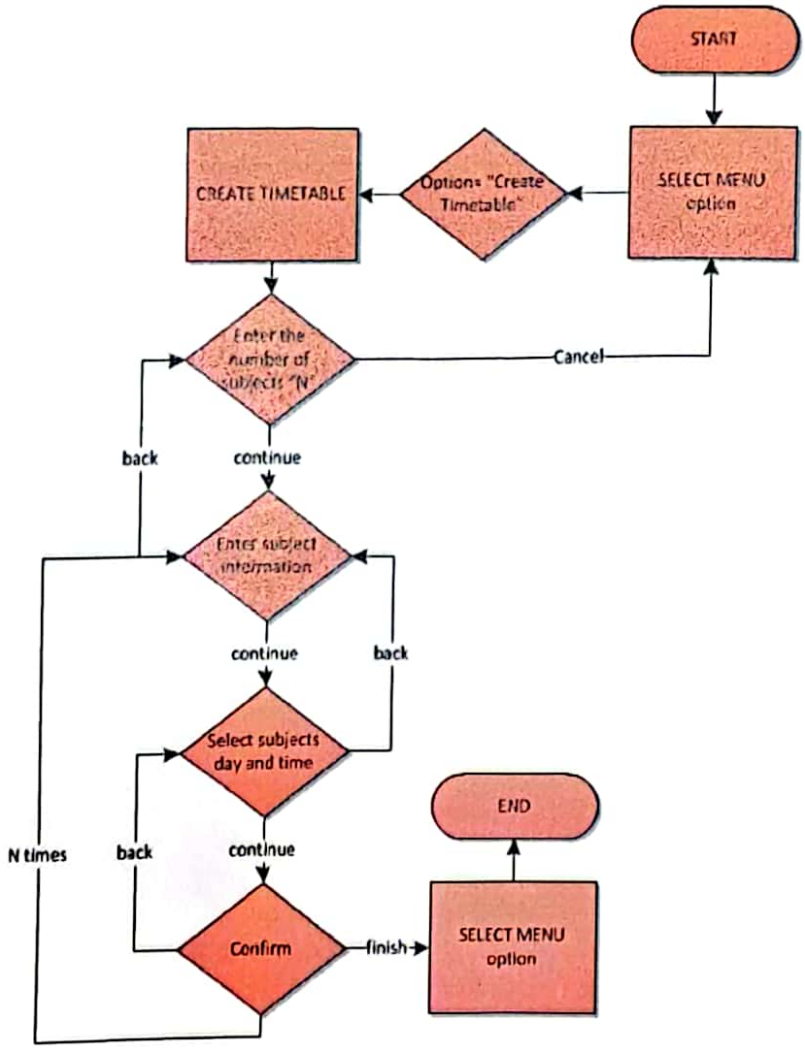


Figure 4.2 Flowchart Create Timetable

4.3 System Design

The system design is divided into three main parts which are data design, database design and interface design.

4.3.1 Data Design

The data of the application is represented using Data Flow Diagrams (DFDs) which are Context Diagram (CD) and Level-0 Diagram and Entity Relationship Diagram (ERD). Three following figures illustrate each diagram correspondingly, **Figure 4.3** CD, **Figure 4.4** Level-0 DFD diagram and **Figure 4.5** illustrates ERD diagram as well.

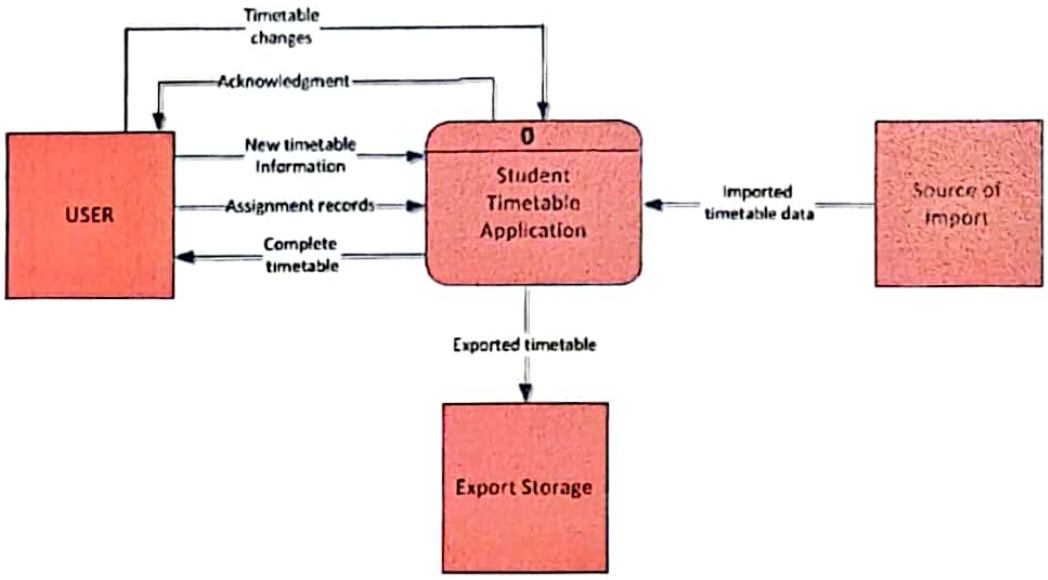


Figure 4.3 Context Diagram

The context diagram illustrated above represents the scope of the application to be developed. The application is operates only with certain class data, in this case is a data related to represent timetable and keep correspondingly assignment records. Also, there is source and destination of imported and exported timetable.

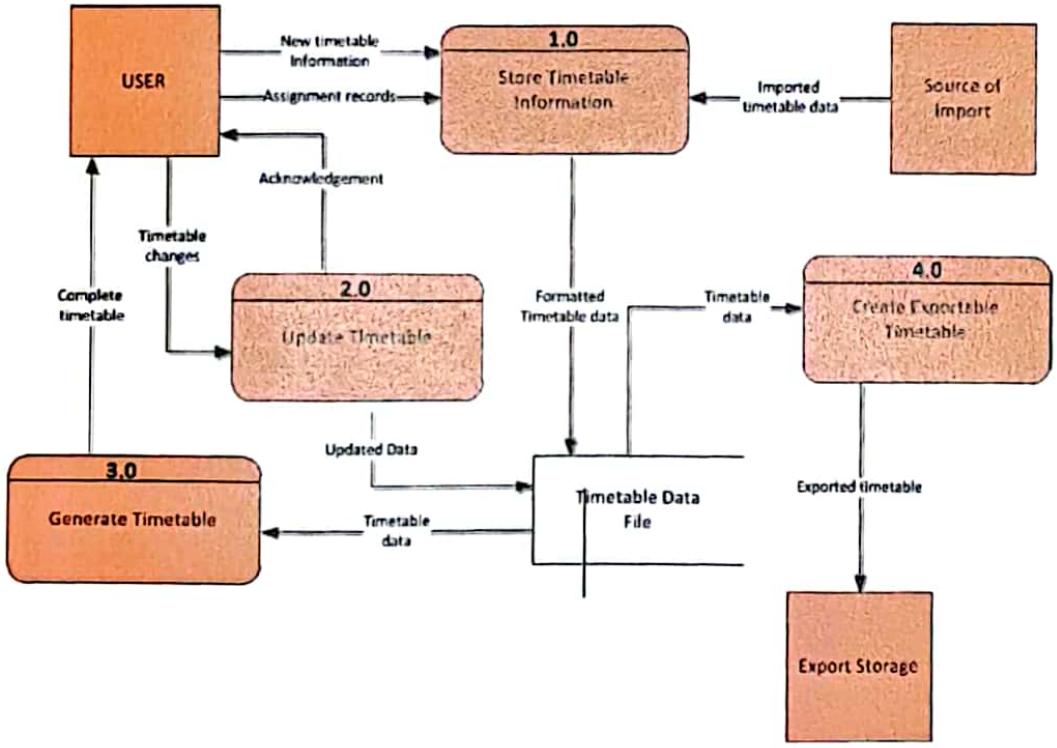


Figure 4.4 Level-0 DFD

The **Figure 4.4** represents Level-0 DFD of the application. As it is shown, there are three entities that interact with the application and four main processes involved. Since the application is going to work only with class data, only one main data storage decided to

be enough. The main attention is given to users, since it is the most intractable part of the program. Users are able to create and make changes to the timetable, as well as to view, import and export it. The timetable data is also able to be exported certain format to the permanent location on the mobile phone or later on, can be sent to the other device. Importing timetable function is one the important functions too. This process is going to be organized by retrieving the timetable data from Uniten system.

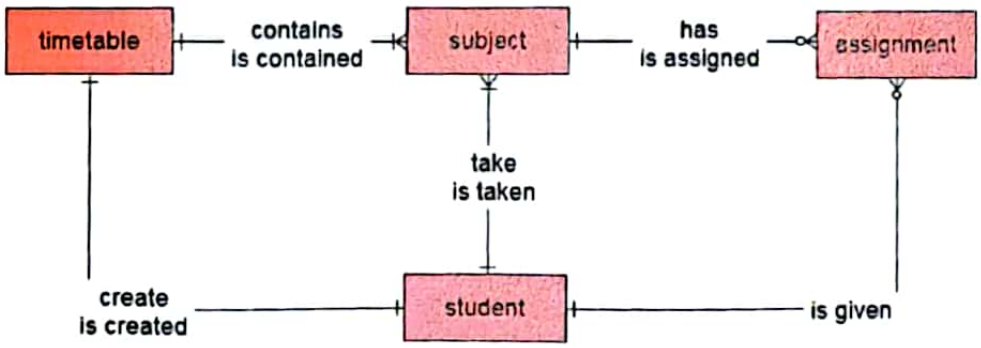


Figure 4.5 ERD

The ERD above in **Figure 4.5** represents the relationship between the main components of the application is organized. As mentioned in earlier section the main figure in the application is user, which in this case is a student. Since each student will use its own application independently, the application is designed such that only one student is able to use certain timetable, in other words the timetable is created individually for each user. Then the rest is obvious that student is able to take one or many subjects as well as be assigned zero or many assignments. The same things go to timetable and subject entities;

such that one timetable can contain one or many subjects and one subject can has zero or many assignments.

4.3.2 Database Design

This section will present all the schemas and tables regarding the database used in the application. These tables give the details of each entity mentioned earlier together with constraints and type of information stored. **Table 4.1** is the main table and consisted of essential class data of the application.

Table 4.1 Timetable Information

Attributes	Data type	Constraint	Content
Subject_code	varchar(7)	Primary key	The code of the subject
Subject_name	varchar(50)	-	The name of the subject
Subject_day	int(1)	-	The day of the subject, numbered.
Subject_time	int(2)	-	Subject time, 24-hour format
Offset_time	int(2)	-	Offset, in case if there is a classes starting not exactly at every hour
Class_venue	varchar(10)	-	The venue of the classes

The table shown above is used to store all the necessary information in order to keep and then properly display student timetable on the screen. Since the application is developed

on mobile phones, it is taken to consideration to have a minimum number table in the database in order to save memory and performance resources. That is why the database does not have the separate table for subject entity, but have a separate table for assignments which is illustrated in **Table 4.2.**

Table 4.2 Assignment Information

Attributes	Data type	Constraint	Content
Assignment_id	int(6)	Primary key	The unique id of each assignment
Subject_code	varchar(7)	Foreign key	The subject of the assignment given
Instruction	varchar(255)	-	A brief description of what was given
Given_date	date()	-	Given date of the assignment
Due_date	date()	-	The due date

An assignment has a subject from which it was given, instructions given by the lecturer and time given to accomplish the task.

Generally, the two tables mentioned above are going to store all the necessary data in order to provide proper functionality of the application.

4.3.3 Interface Design

Figure 4.6 until **Figure 4.13** roughly demonstrates the essential points of the future application user interface.

Figure 4.6 illustrates the main menu interface of the application.

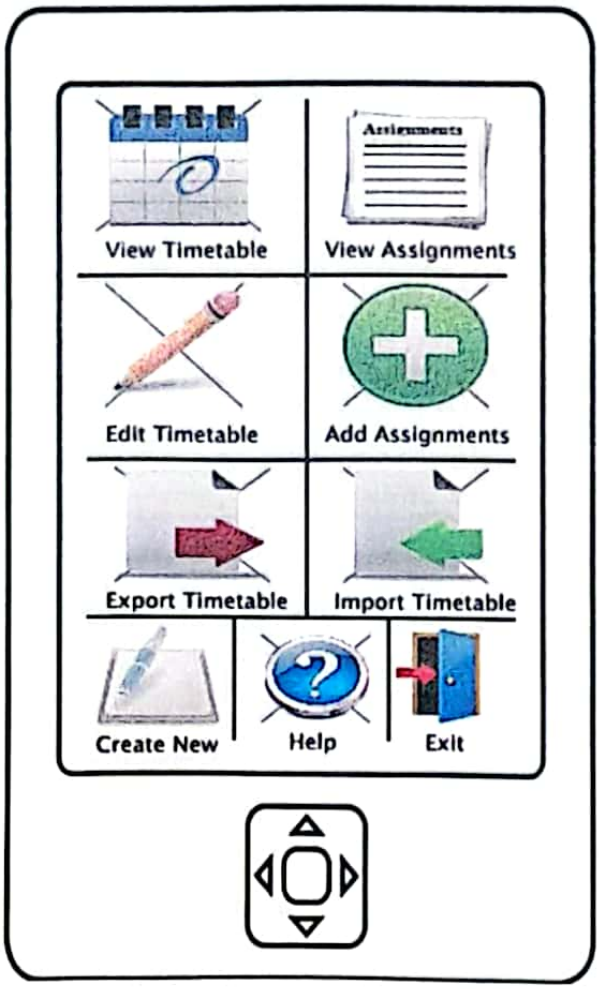


Figure 4.6 Main Menu UI

As it is shown on the figure above, the main menu consists of nine option where each of them has it is own functionality. Users are able to choose one of this option and continue using the application depending on their needs.

The next **Figure 4.7** demonstrates the view timetable interface.

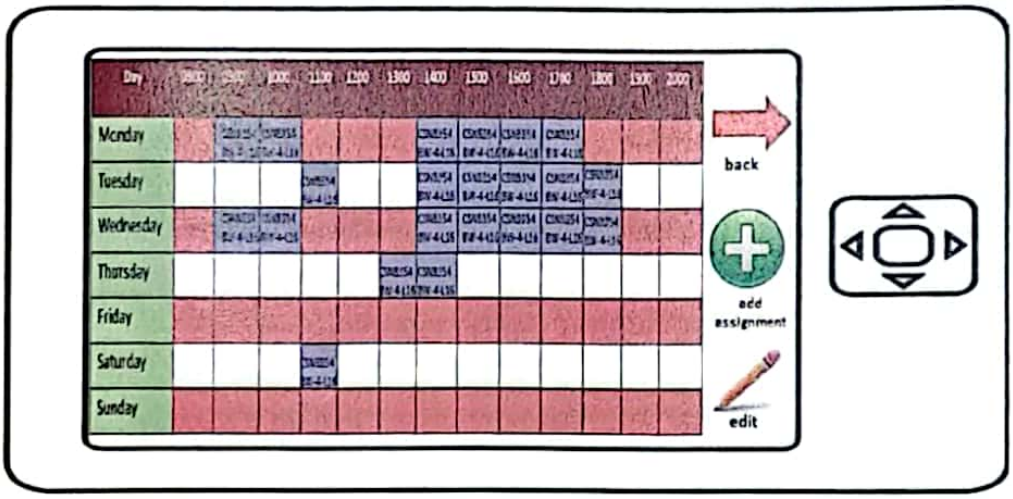


Figure 4.7 View timetable interface

In view timetable option users are able actually to view their timetable, but there are also abilities to add assignment or edit timetable.

Figure 4.8 below illustrates view assignments menu interface.

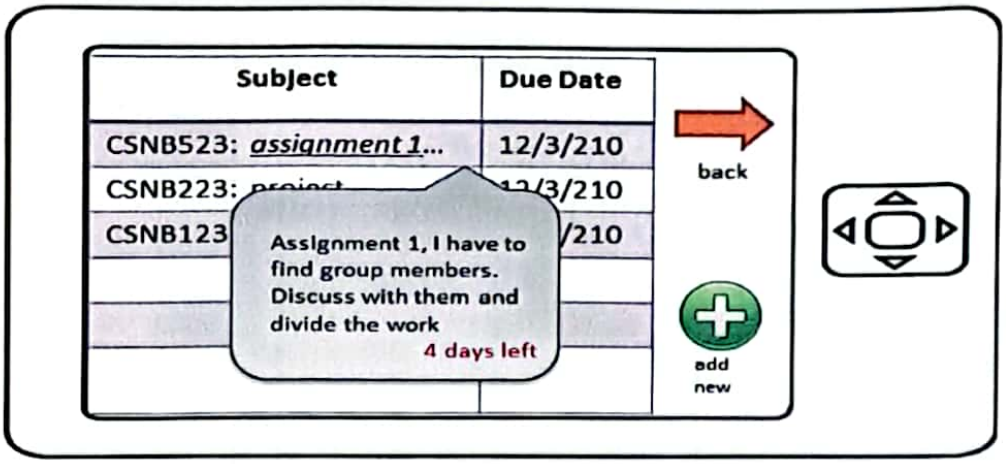


Figure 4.8 View assignments interface

In the view assignments option, all assignments are listed in the table with corresponding date of assignment submission. By simply clicking on the necessary assignment users are able to view its details.

The next **Figure 4.9** shows the add assignment option interface.

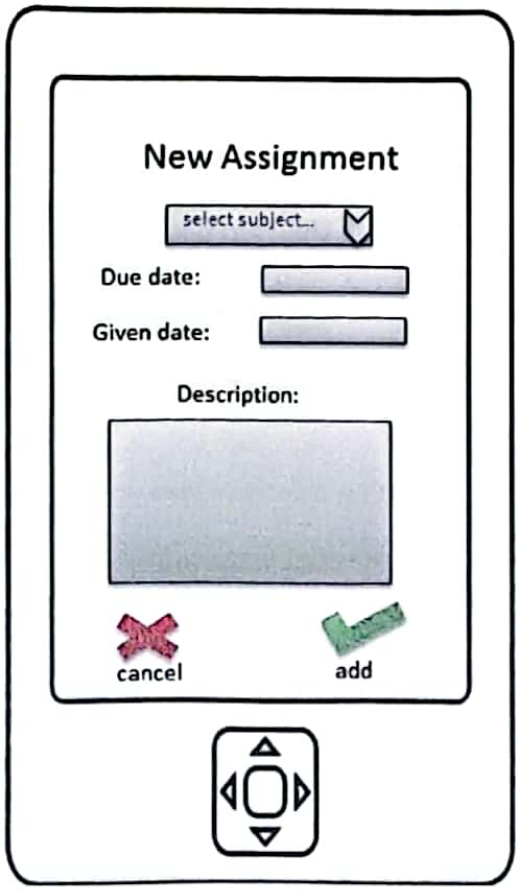


Figure 4.9 Add assignment interface

In this figure illustrated above, users are able to add new assignment in order to keep record of and later, whenever it is necessary to refer back to it.

The following several figures roughly illustrate process of creating timetable. Note, that create timetable function will overwrite previous timetable data. **Figure 4.10** is the first step of timetable creation.

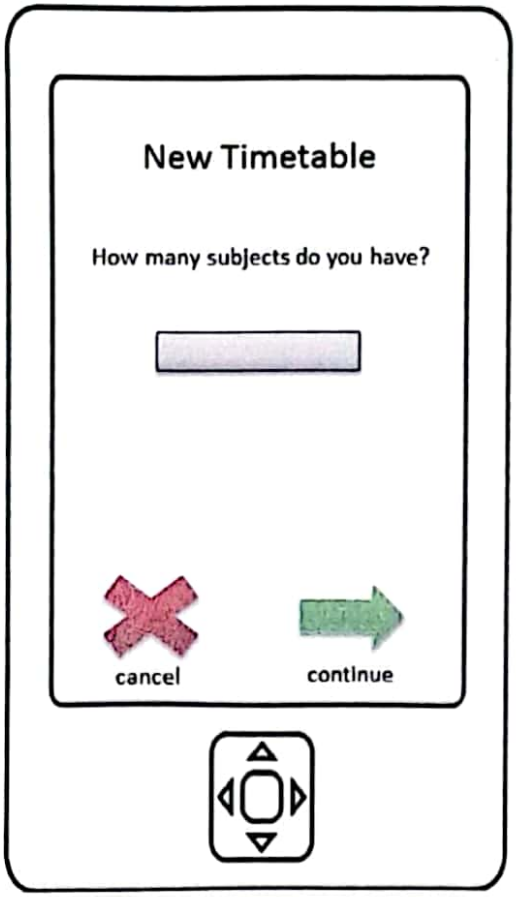


Figure 4.10 Timetable creation interface step 1

The figure above prompts to enter the number of subjects that user registered in current semester. The next step is illustrated in **Figure 4.11** where user has to enter information about subjects in sequential order.

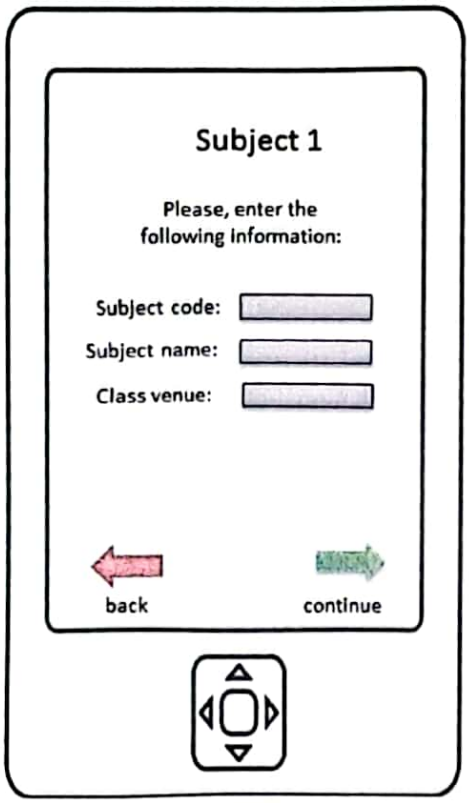


Figure 4.11 Timetable creation interface step 2

In the figure above program prompts user to enter basic subject information such as subject name, subject code and venue. **Figure 4.12** illustrates the interface where user has to tick the time and day of the subject.

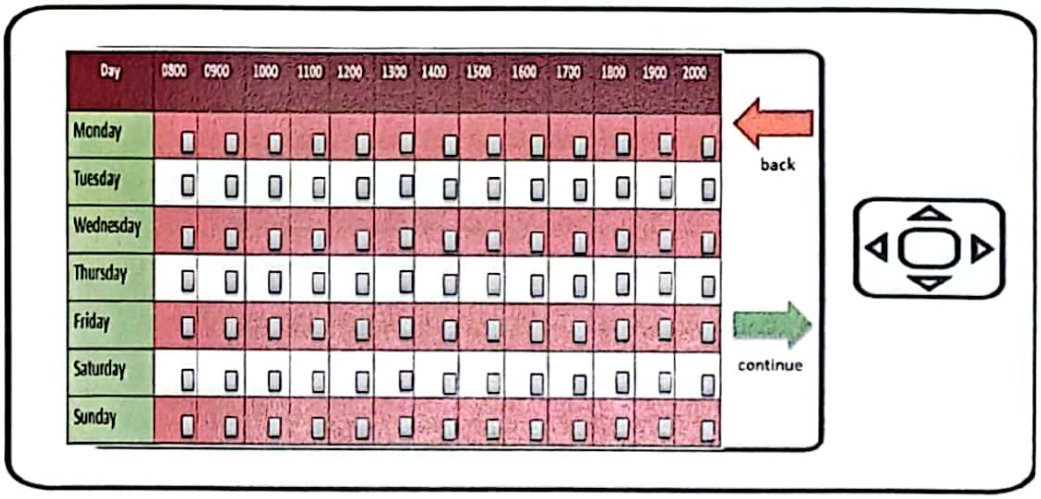


Figure 4.12 Timetable creation interface step 3

As illustrated in **Figure 4.12,** user is able to simply tick checkboxes in order to assign the time and day of the subject. After pressing continue button the subject is added into time table as shown in **Figure 4.13.**

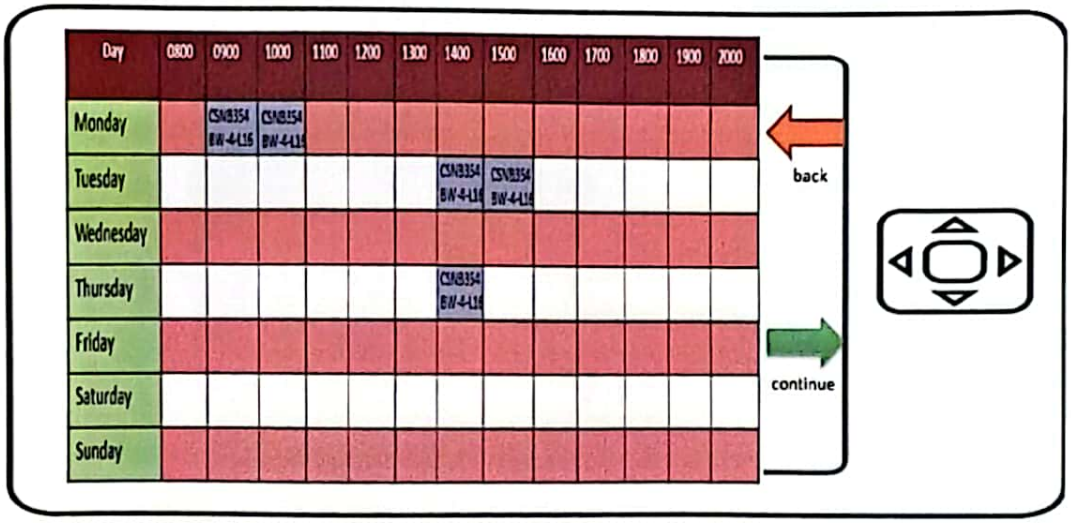


Figure 4.13 Timetable creation interface step 4

The figure above is the last step in the whole cycle of timetable creation process. These steps continue depending on the number of subjects.

All the figures illustrated in this section are sample interfaces, during the actual coding and designing the real interface some slight changes might take place.

CHAPTER 5

IMPLEMENTATION

5.1 Physical Design

The implemented application on Android platform consists of total number of eight main activities. They are: main menu, view timetable, edit timetable, create timetable, view assignments, add assignments, import timetable and options. Each activity performs certain function. In this section all of the pages of the application mentioned above are going to be described in detail.

Main menu, as the name says, is invoked when users first opens the application. Below, it is the figure illustrating this activity.

Figure 5.1 Main menu

All other activities illustrated as icons are placed on this menu and users are able to access them by simply clicking on each of them.

The view timetable activity consists of several buttons and timetable itself. The entire scree almost occupied by the timetable, which is similar to Uniten online class timetable.

On the bottom of the screen group of four buttons are located. The back button returns application to the main menu. The add assignment button opens the corresponding activity in order to add new assignment. The edit button opens another activity which lets user to edit the time, day and the venue of the timetable. Make screenshot button allows user to take a snapshot of the current timetable and saves it to the phone. This button plays the role of the export timetable function. **Figure 5.2** illustrates the view timetable activity.

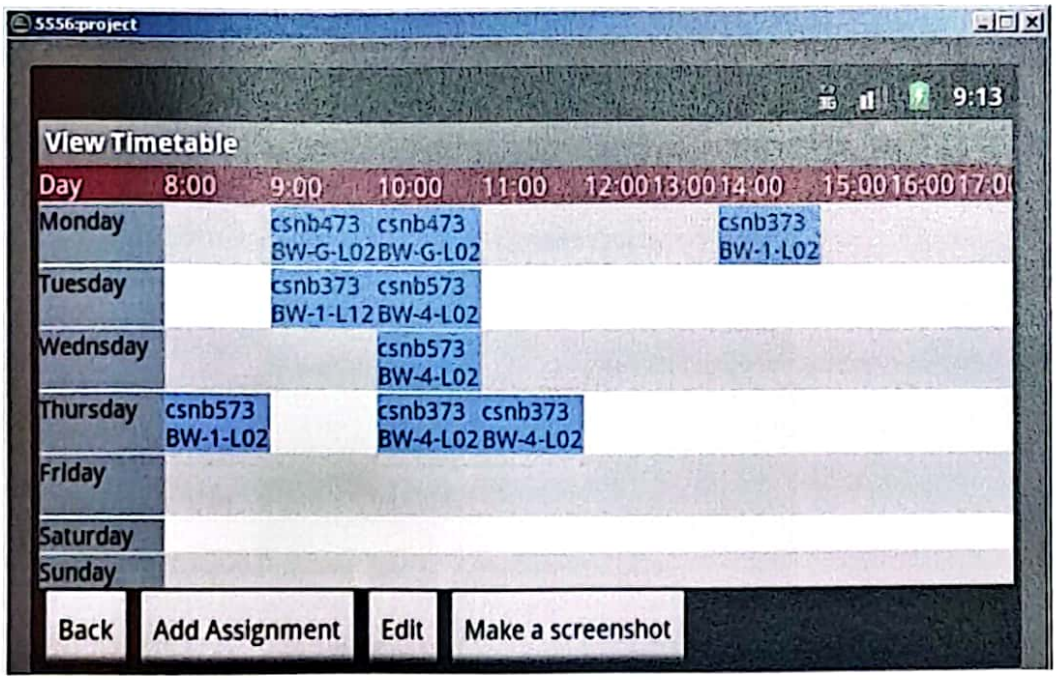


Figure 5.2 View timetable

Edit timetable activity, as was mentioned earlier in this section allows to user to edit time, day and venue of the class. It contains a form which consists of several views such as subject spinner, the day and time to be edited, new day and time and the new venue of the class. After the user has specified correct information, user should to press edit button, which will verify the entered information and if valid, the application will edit the timetable. The total number of five spinners provides better usability to user and same as in other activities it has back button. **Figure 5.3** demonstrates the edit timetable activity.

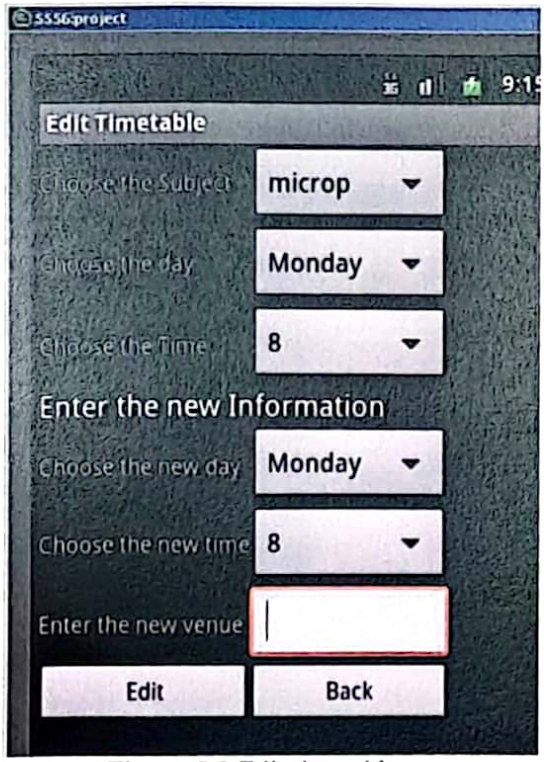


Figure 5.3 Edit timetable

One of the most involved parts of the application development was the create timetable function, because it consists of four steps. Once, user press to the icon, a dialog box will appear telling that, creation the new timetable will delete the previous one, which means that users are able to keep only one timetable at a time. It is done so, in order to prevent wasting of resources on mobile phone. After user confirmation, the first step asks to enter number of subjects that user would like to have in the new timetable. Further, application prompts user to enter the information about the first class as subject name and code. After that user just tick the checkboxes about the time and day of the class and enters the venue for every checked class. The last step is to confirm the data entered. If users want to redo the step he/she is able to do it by pressing cancel button. Otherwise, user required to press next button, to proceed with the next subject. These steps are continued as many times as the number of classes which user has entered. **Figures 5.4 – 4.9** illustrates all steps which are involved in timetable creation in details.

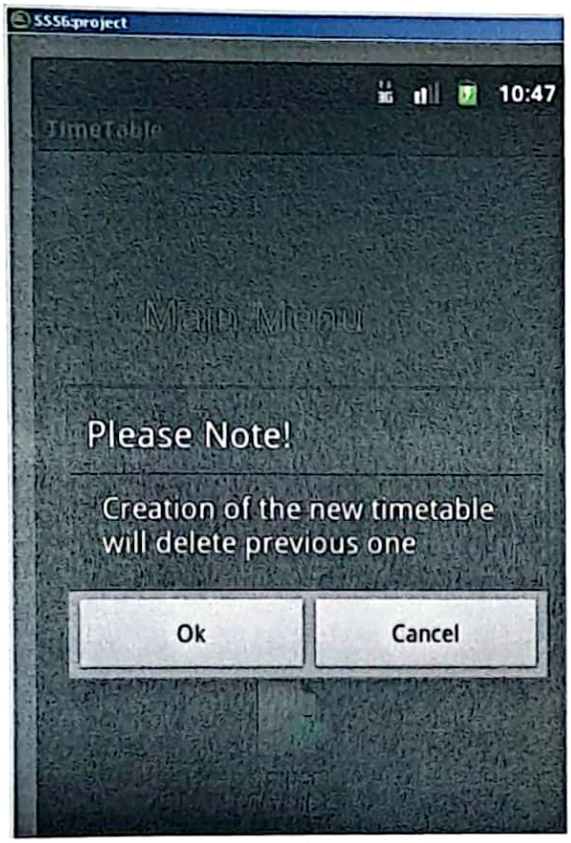


Figure 5.4 User notification

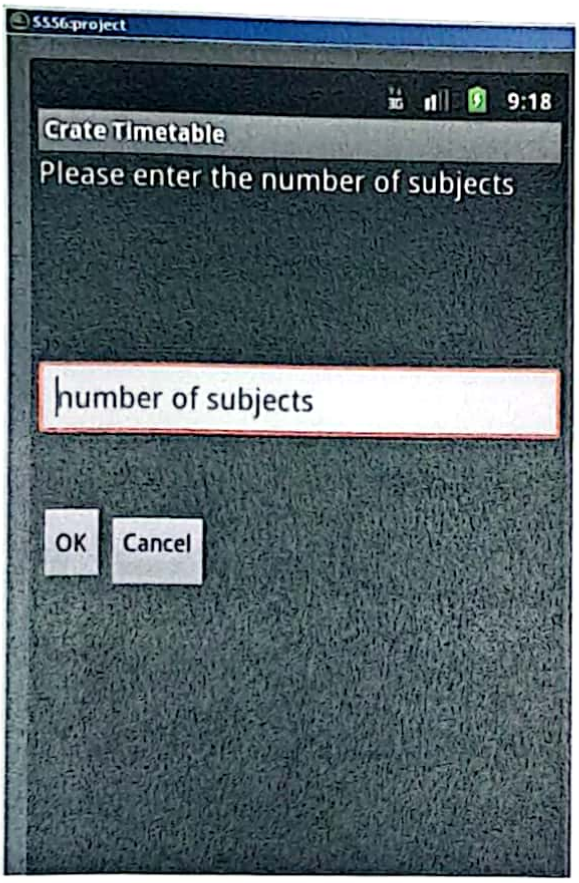


Figure 5.5 Create timetable step 1

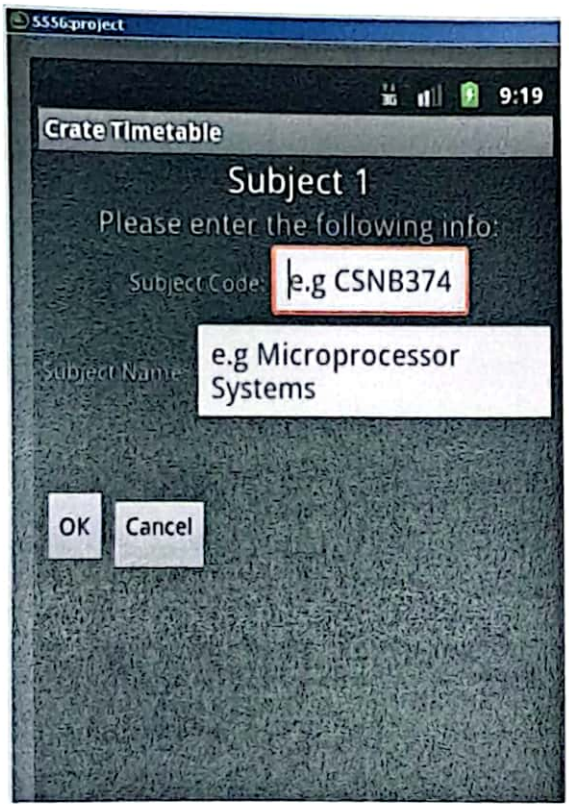


Figure 5.6 Create timetable step 2

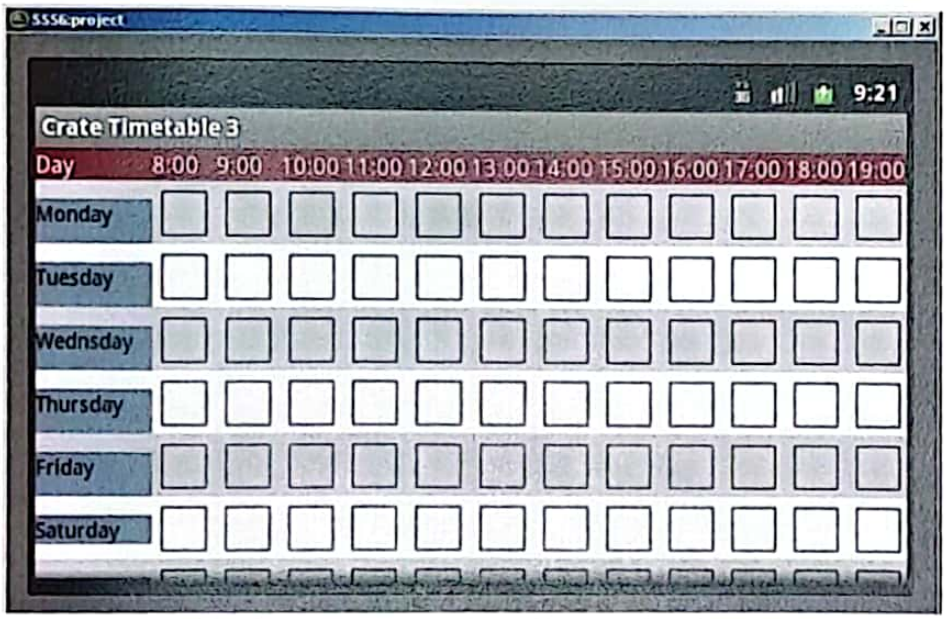


Figure 5.7 Create timetable step 3

Figure 5.8 Entering venue information

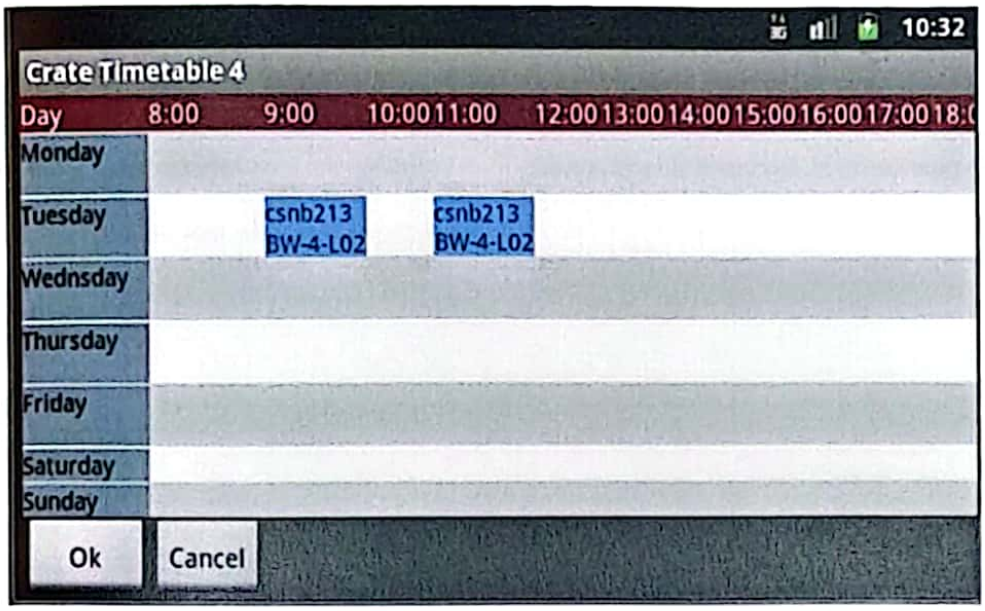


Figure 5.9 Confirmation of entered data

As another objective of this project is that the application allows users to keep their class assignments. Add assignment function allows user to add assignment, which asks user to enter the subject on which the assignment given, due date and short description of what to do. All added assignments are accessible from view assignments activity. In the view assignments activity, users are able to see all the assignments together with the following information:

- The subject on which the assignment has given;
- Assignment description;
- The due date;
- Number of days left to finish the assignment.

In case of assignment due date expiration, the due date label becomes red. Next to due date, checkboxes are placed in order to allow to user to delete assignments. Users are able to delete both expired and unexpired ones. **Figure 5.10** and **Figure 5.11** illustrate both, view assignment and add assignment pages.

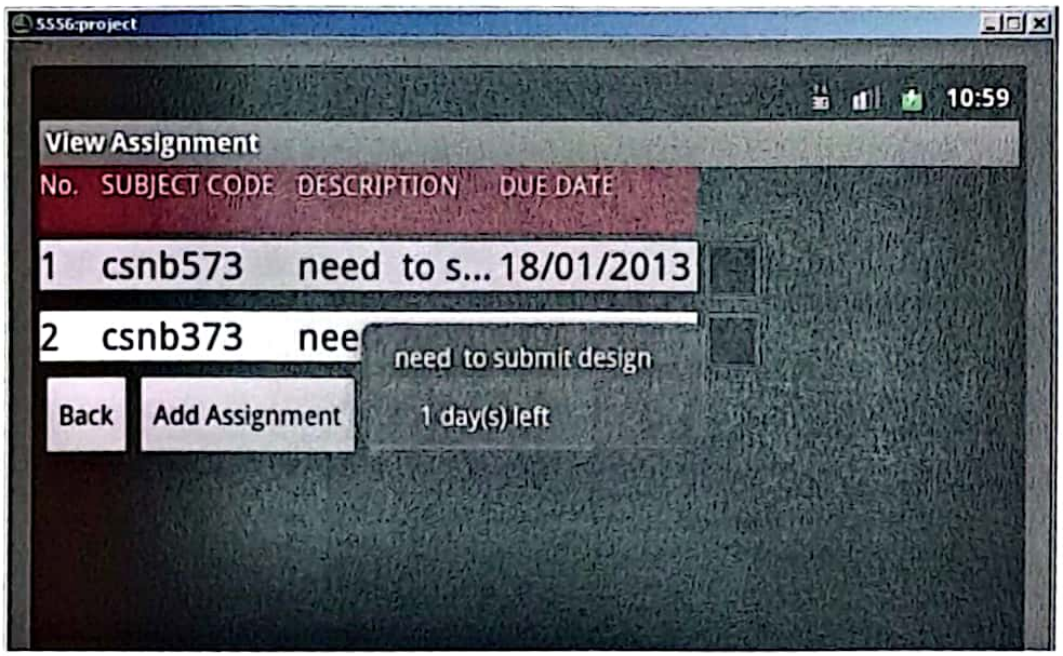


Figure 5.10 View assignments

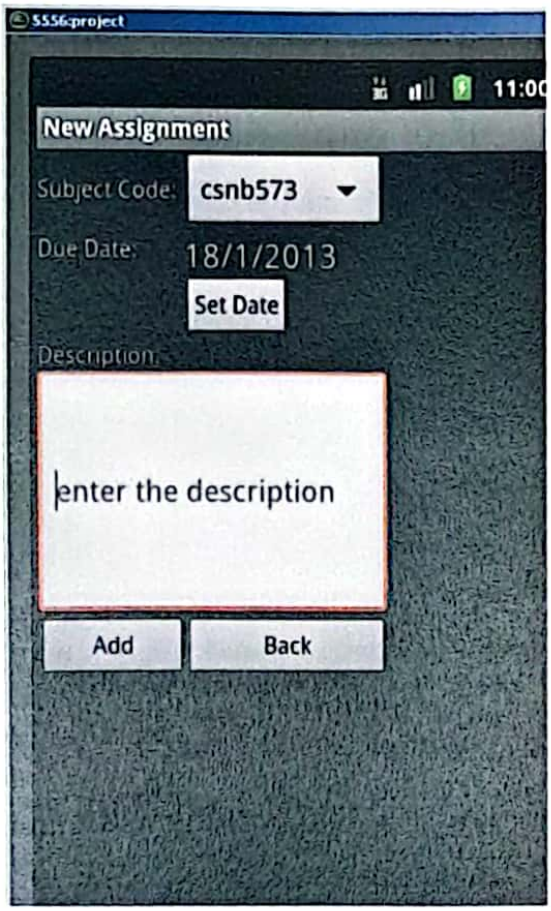


Figure 5.11 Add assignments

The option activity allows user to enable/disable auto-silent mode and assignment notification features. Both functions are implemented using background threads, which are able to send notification and put the phone in silent mode from background. **Figure 5.12** shows the options activity.

Figure 5.12 Options

The last, but not least is the import timetable function of the application. The actual processing of the activity is totally hidden from the user. But in the background, in case of active network connection, this activity connects to the apache server, forces it to extract timetable data to the .txt file and the activity retrieves this information by using java streams. Information retrieved then stored to the database and appears to the user as a normal timetable. As in case of timetable creation, activity informs that it deletes the previous timetable data. Also, the corresponding message appears if there is no active network connection.

5.2 Technical Design

In order to develop the application as the main platform Eclipse IDE together with Android Development Tools plugin were used. Since the Android platform is developed on Java, the main programming language is used was Java including all its standard libraries and functions. Also, Android platform depending on the API level provides its own libraries. Those libraries provide users with wide variety of functionalities as designing the user interface, all on-click and on-touch events inter-activity communication and so on. In this case Android 2.3.3 and API level 10 are used. Also, JodaTime 2.1 additional library is added in order to work with due date of the assignments. In order to keep the timetable and assignments data SQLite database is used.

5.3 Coding

The most important parts of the coding are the storing properly the timetable data into the SQLite database and properly display it so that it appears to the user as a normal timetable. Some important code snippets are provided in this section.

5.3.1 Database Handler

The database handler class is implemented in order to handle all the processes regarding the creation, insertion, updating and deletion of data. Android API provides database

helper class in order to simplify the database implementation of application development. Below is the example of data insertion into database in Android which is used in current application.

```
/**
 * All CRUD(Create, Read, Update, Delete) Operations
 */

//Insert data
public boolean insertValues( String name, String code, String venue, int time, int day) {

    SQLiteDatabase db = null;
    try{
        db = this.getWritableDatabase();

        ContentValues values = new ContentValues();

        values.put(KEY_DAY, day);
        values.put(KEY_TIME, time);
        values.put(KEY_SUBJECT_NAME, name);
        values.put(KEY_SUBJECT_CODE, code);
        values.put(KEY_SUBJECT_VENUE, venue);

        long result = db.insertOrThrow(TABLE_NAME, null, values);
           return result > 0;


    }catch (SQLException ex) {
           Log.w("SQLException", ex.fillInStackTrace());
           return false;
    }finally {
          if (db != null) {
              db.close();
          }
     }
}
```

Figure 5.13 Insert method in Database Handler class

5.3.2 Timetable representation

In order to implement the actual representation of the timetable to the screen, first of all, the corresponding java class must extends Android Activity class, which mandatory for any kind of class that is going to work with android device screen. After extending

Activity class developer must override some native methods, which are declared in the abstract class. One of the such method is OnCreate() method which is invoked when an activity called. Inside this activity programmer specifies the link to the layout which is written using XML and responsible for the appearance of the activity. Each activity must have one XML layout.

In this case view_timetable.xml layout is used in order to represent the timetable and its components. **Figure 5.14** illustrates the snippet of XML file.

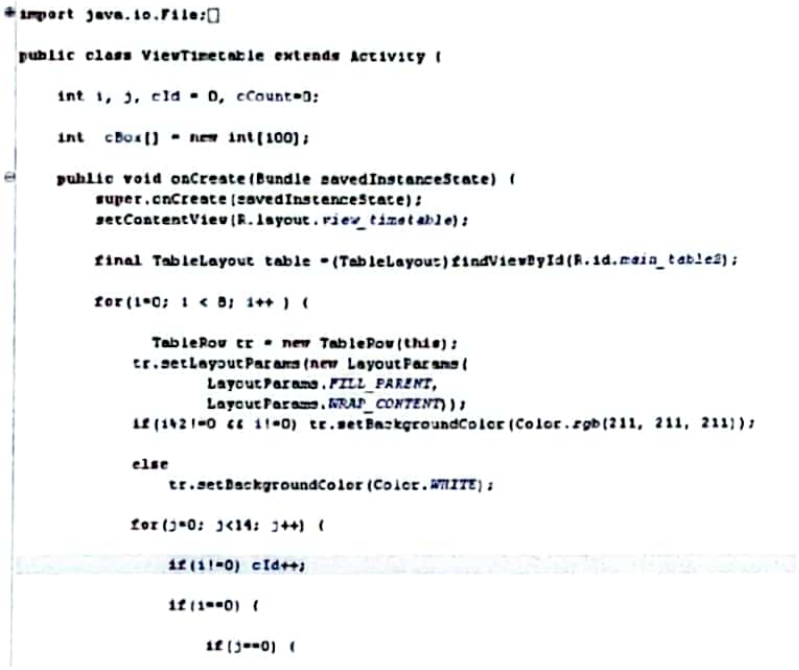
```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent" android:layout_height="fill_parent"
    android:scrollbars="vertical">

<TableLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
     android:id="@+id/main_table2"
     android:layout_height="wrap_content"
     android:layout_width="fill_parent"
     android:stretchColumns="1"
     android:scrollbars="vertical"
     >

</TableLayout>
</ScrollView>
```

Figure 5.14 View Timetable XML

The java class to which this XML layout is linked is shown in **Figure 5.15**.



```
import java.io.File;

public class ViewTimetable extends Activity {

    int i, j, cId = 0, cCount=0;

    int  cBox[] = new int[100];

    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.view_timetable);

        final TableLayout table =(TableLayout)findViewById(R.id.main_table2);

        for(i=0; i < 8; i++ ) {

              TableRow tr = new TableRow(this);
            tr.setLayoutParams(new LayoutParams(
                    LayoutParams.FILL_PARENT,
                    LayoutParams.WRAP_CONTENT));
            if(i%2!=0 && i!=0) tr.setBackgroundColor(Color.rgb(211, 211, 211));

            else
                tr.setBackgroundColor(Color.WHITE);

            for(j=0; j<14; j++) {

                if(i!=0) cId++;

                if(i==0) {

                    if(j==0) {
```

Figure 5.15 View Timetable java class

As it is shown in the figure above, as main algorithm two for-loops are implemented in order to represent the main timetable. One loop is for day column another one is for time rows. On every step, the check is done, in order to identify whether there is class on that time and day or not. If the data in the database corresponds to the row and column, it gets represented in that cell.

5.3.3 Implementation of inter-activity communication

Since the timetable application has a lot of interchanging of the activities and information between them, Android API provides library called Intents. Intents are used to provide

communication between processes including moving from one activity to another, passing information and so on. Below, the code illustrates an example of intents used in the application.

```
Intent intent = new Intent(CreateStepThird.this, CreateStepFourth.class);
intent.putExtra("Day", day);
intent.putExtra("Time", time);
intent.putExtra("Count", numberOfClasses[0]);
intent.putExtra("SubjectName", subjectNameStr);
intent.putExtra("SubjectCode", subjectCodeStr);
intent.putExtra("SubjectVenue", subjectVenueStr);
startActivity(intent);
finish();
```

Figure 5.16 Intents in Android

5.3.4 Android native widgets

Android API is very reach and has a lot of widgets in order to support usable user interface. One of the most frequently widget was displaying the short messages to inform user called Toasts. **Figure 5.17** demonstrates the implementation of toasts in Android.

```
Toast toast;
toast = Toast.makeText(MainActivity.this, "Timetable imported!", Toast.LENGTH_SHORT);
toast.setGravity(Gravity.CENTER, 0, 0);
toast.show();
```

Figure 5.17 Toasts in Android

And **Figure 5.18** illustrates the toast in action.

Figure 5.4 User notification

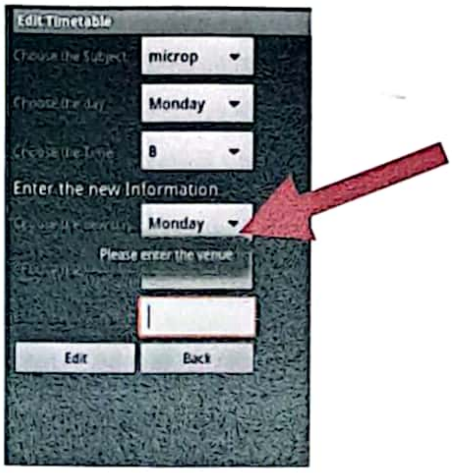


Figure 5.18 Toast in action

Another widget called alert boxes in Android allows not only inform user, but also take appropriate action. Below is the snippet which allows creating an alert box an set positive and negative button actions.

```
final AlertDialog.Builder alert = new AlertDialog.Builder(this);

alert.setPositiveButton("OK", new DialogInterface.OnClickListener() {
public void onClick(DialogInterface dialog, int whichButton) {

    dbHelper2.removeAll();
    Intent intent = new Intent(MainActivity.this, CreateStepFirst.class);
    startActivity(intent);
  //  finish();
}
});

alert.setNegativeButton("Cancel", new DialogInterface.OnClickListener() {
  public void onClick(DialogInterface dialog, int whichButton) {
    // Canceled.

  }
});
```

Figure 5.19 Alert-box implementation

And **Figure 5.20** illustrates the alert box in action.

Figure 5.20 Alert-box in action

As it was mentioned earlier one of the features of the developed application is to send notification about coming assignments. In this case, Android API provides special widget called notification manager. **Figure 5.21** illustrates the code that creates the notification.

```
//Notification Manager

String ns = Context.NOTIFICATION_SERVICE;
mNotificationManager = (NotificationManager) getSystemService(ns);
String message="You have unfinished assignments, please, refer to your timetable applications";
String title="Upcoming/expired assignments!";
int icon = R.drawable.create_time;
CharSequence tickerText = message;
long when = System.currentTimeMillis();
notification = new Notification(icon, tickerText, when);
Context context = getApplicationContext();
CharSequence contentTitle = title;
CharSequence contentText = message;
Intent notificationIntent = new Intent(this, BaseActivity.class);
PendingIntent contentIntent = PendingIntent.getActivity(this, 0, notificationIntent, 0);
notification.flags = Notification.FLAG_AUTO_CANCEL;
notification.setLatestEventInfo(context, contentTitle, contentText, contentIntent);

notify.setName("notify");
dbHelper1 = new DatabaseHandler(this);
```

Figure 5.21 Notification Manager

Another important feature of the developed application is to put the device into silent mode in case there is a class on that time. **Figure 5.22** is a snippet that puts the device into silent mode.

```
//Putting device in silent mode
if(found) {
    if(audio_mngr.getRingerMode()==AudioManager.RINGER_MODE_VIBRATE || audio_mngr.getRingerMode()==AudioManager.RINGER_MODE_NORMAL)
        audio_mngr .setRingerMode(AudioManager.RINGER_MODE_SILENT);
}
```

Figure 5.22 Auto-silent mode implementation

As it was mentioned earlier these two features are implemented inside the background threads. The way to implement threads is by using simple java Thread class.

Spinners are also one of the types of widgets that Android API provides. Spinners do not differ from familiar to us drop-down lists. **Figure 5.23** gives an example of the spinner implementation.

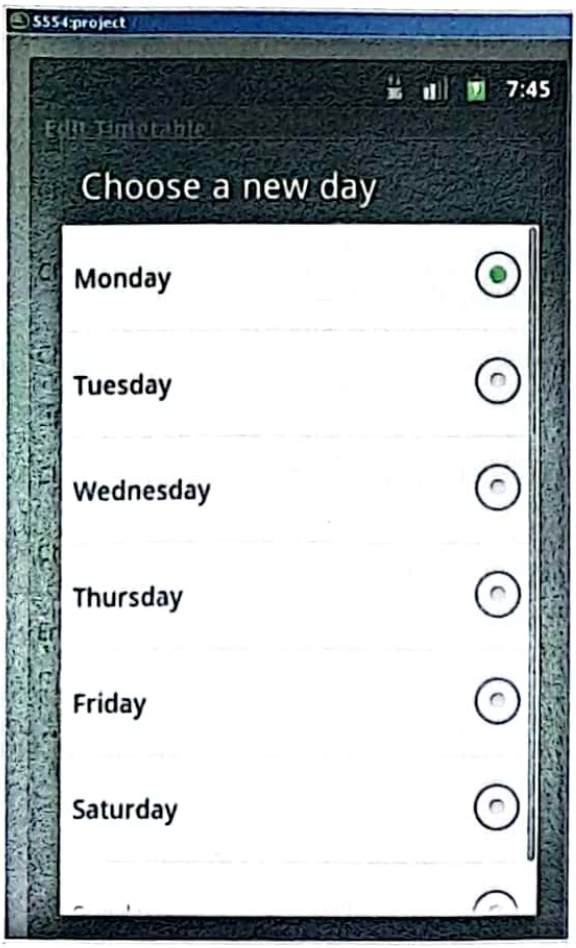


Figure 5.23 Spinner view

Date picker is widget supported by Android API in order to give opportunity to user to set the date, rather than inputting it manually. **Figure 5.24** is an example of the date picker used the application.

Figure 5.24 Date picker view

5.4 Timetable export and import

Exporting timetable feature of the application is implemented by simply using bitmap compression and java streams. The idea is to take a root view of the activity, write it to a file by using output stream and bitmap compression as a JPEG file. **Figure 5.25** is a code snippet that demonstrates screenshot implementation.

```
boolean success = false;

String mPath = Environment.getExternalStorageDirectory().toString()+ "/myFile.png";

Bitmap bitmap;

View v1 = table.getRootView();
v1.setDrawingCacheEnabled(true);
bitmap = Bitmap.createBitmap(v1.getDrawingCache());
v1.setDrawingCacheEnabled(false);

OutputStream fout = null;
File imageFile = new File(mPath);
  try
    {
        imageFile.createNewFile();
        fout = new FileOutputStream(imageFile);
        bitmap.compress(Bitmap.CompressFormat.JPEG, 90, fout);
        fout.flush();
        fout.close();
        success=true;
    }
```

Figure 5.25 Export timetable

Since, importing timetable happens from remote site, this function involves the following steps:

- Retrieve the data from remote site;
- Store the retrieved data in a proper form into database;
- Represent the imported data as a timetable.

Retrieving data from the remote site is implemented by using HTTP client on the device, which connects to the apache server, and the server has script which downloads the data from the database to certain location as a .txt file. After that, normal java IO streams are used in order to read the data from the remote source. The data that has been read is stored into the database in certain format, from which timetable is represented to the user. **Figure 5.26** is a code snippet that shows the implementation of the timetable import.

```
        String myurl = "http://10.0.2.2/schedule/writeAndDownload.php";

        HttpClient httpclient = new DefaultHttpClient();
            HttpResponse response = httpclient.execute(new HttpGet(myurl));
            StatusLine statusLine = response.getStatusLine();
            if(statusLine.getStatusCode() == HttpStatus.SC_OK){
                ByteArrayOutputStream out = new ByteArrayOutputStream();
                response.getEntity().writeTo(out);
                out.close();
                String responseString = out.toString();
                //..more logic
            } else{
                //Closes the connection.
                response.getEntity().getContent().close();
                throw new IOException(statusLine.getReasonPhrase());
            }

        }catch(Exception e){

            e.printStackTrace();
        }

        String nextLine;
        URL urlFile = null;
        URLConnection urlConn = null;
        InputStreamReader  inStream = null;
        BufferedReader buff = null;

   urlFile  = new URL("http://10.0.2.2/schedule/data.txt" );
   urlConn = urlFile.openConnection();
  inStream = new InputStreamReader(
                  urlConn.getInputStream());
    buff= new BufferedReader(inStream);

// Read and print the lines from index.html
 while (true){
     nextLine =buff.readLine();
     if (nextLine !=null){

 //tokenize and put into db

        StringTokenizer tkns = new StringTokenizer(nextLine);


        dbHelper2.insertValues(tkns.nextToken().toString(),
                tkns.nextToken().toString(),tkns.nextToken().toString(), Integer.parseInt(tkns.nextToken().toString()),
                Integer.parseInt(tkns.nextToken().toString()));
```

Figure 5.26 Import timetable

CHAPTER 6

TESTING AND VERIFICATION

6.1 Introduction to application testing

Testing is a process of debugging application in order to find errors or improper application behavior. There are three types of testing used, which are unit testing, usability testing and user acceptance testing.

6.2 Unit testing

Unit testing is a type of test focused on each function of the application in order to check it on correctness of input and error prevention. Unit testing consist of certain amount of test cases applied to each function.

6.2.1 Create timetable function

Test objective: To test all steps involved in timetable creation process.

Table 6.1 Create timetable test case

Test Case	Step	Test data	Expected Result	Actual Result
1	Step 1. Entering empty number of subjects	Number of subjects = ""	Should display message= "Please, enter the number"	Displays message= "Please, enter the number"
2	Step 1. Entering number higher than 10	Number of Subjects = 0 or > 10	Should display message= "Number should not be more than 10 or equal to 0"	Displays message= "Number should not be more than 10 or equal to 0"
3	Step 1. Entering number of subjects between 0 and 10	Number of subjects !=0 and <10	Should go to the second step	Goes to the second step
4	Step 2. Entering empty subject name or subject	Subject name = "" or subject code	Should display message =	Displays message =

	code	= ""	"Please enter the subject name" or "Please, enter the subject code"	"Please enter the subject name" or "Please, enter the subject code"
5	Step 2. Entering subject name and code	Subject name="Advanced OS", subject code= "CSNB334"	Should go to step 3	Goes to step 3
6	Step 3. Not checking at least one checkbox	Number of checkboxes checked = 0	Should display message = "Please, please check at least one checkbox"	Displays message = "Please, please check at least one checkbox"
7	Step 3. Checking more than 10 checkboxes	Number of checkboxes > 10	Should display message = "Maximum 10 checkboxes can be checked"	Display message = "Maximum 10 checkboxes can be checked"
8	Step 3. Checking	Number of	Should display	Displays dialog

	appropriate number of checkboxes	checkboxes !=0 and <10	dialog box asking to enter class venues	box asking to enter class venues
9	Step 3. Entering values into dialog boxes	Class names	Should go to step 4 and print entered data into timetable	Goes to step 4 and prints timetable data

6.2.2 Edit time table function

Test objective: To test the correctness of timetable edits.

Table 6.2 Edit timetable test case

Test Case	Step	Test data	Expected Result	Actual Result
1	Choosing not corresponding subject to the time and day in the timetable	Subject code = "csn473" , time = 9, day = Monday	Should display message = "Subject to be edited doesn't exist"	Displays message = "Subject to be edited doesn't exist"

2	Entering new time and day with clash	New time = 10, new day = "Monday"	Should display message = "Clashing between subjects"	Displays message = "Clashing between subjects"
3	Entering valid information	Subject code = "csn573" , time = 9, day = Monday, New time = 11, new day = "Monday"	Should display message = "Update successful" and update the timetable	Displays message = "Update successful" and updates the timetable

6.2.3 Add new assignment function

Test objective: To test the correctness of adding new class assignment.

Table 6.3 Add new assignment test case

Test Case	Step	Test data	Expected Result	Actual Result

1	Entering the empty description	Description = ""	Should display message = "Please, enter subject description"	Displays message = "Please, enter subject description"
2	Setting expired date	Set date before today's date	Should display message = "Due date already expired"	Displays message = "Due date already expired"
3	Entering description and setting valid date	Description = "Need to submit proposal" and due date = 25/Jan/2013	Should display message = "Assignment successfully added" and adds assignment into database	Displays message = "Assignment successfully added" and adds assignment into database

6.2.4 Import timetable function

Test objective: To test the correctness of importing new timetable.

Table 6.4 Import timetable test case

Test Case	Step	Test data	Expected Result	Actual Result
1	Pressing import icon without network connection	No active network connection	Should display message = "Sorry, no network connection"	Displays message = "Sorry, no network connection"
2	If any exception occurs e.g. Server unavailable, IO Exception etc.	Exception occurs	Should display message = "Sorry, an error occurred, please, try later"	Displays message = "Sorry, an error occurred, please, try later
3	Active network connection and no exceptions	Active network connection and no exceptions	Should display message = "Timetable imported" and go to view timetable	Displays message = "Timetable imported" and goes to view timetable

6.3 Usability testing

The purpose of the usability testing is to determine whether the system is easy to use by user, properly designed and error preventive.

6.3.1 Heuristic evaluation

Heuristic evaluation is a technique that helps to evaluate user interface design problems. Generally, there are number of principles to which, user interface should correspond. **Table 6.5** is a summary of heuristic evaluation, which shows the application satisfaction.

Table 6.5 Heuristic evaluation

No.	Heuristic evaluation principle	Yes	No	Comment
1	Visibility of system status	√		Application always provides feedback when an action is done. E.g. if assignment added, timetable edited, imported or exported
2	Match between system and real world	√		The icons of the menu correspond to those used in real life. E.g. logo of timetable, assignments and so on.

3	User control and freedom	√		Users are always able to exit from any step by using android's in-built back and home button.
4	Consistency and standards	√		Application buttons and icons are all in same standard
5	Error prevention	√		Application prevents user from entering wrong data by validating user input
6	Recognition rather than recall		√	User must remember the classes that were picked, classes to be changed
7	Flexibility and efficiency of use	√		Application can be easily used by inexperienced and experienced users
8	Aesthetic and minimalist design	√		Application doesn't have redundant information
9	Help users recognize, diagnose and recover from errors	√		Application guides users in order to make valid input
10	Help and documentation		√	Application doesn't provide help or documentation

6.4 Integration testing

The main goal of this type of test is to group together certain parts and ensure that application is correctly functioning as whole system. Unfortunately, no tools are applied in order to conduct integration testing. The test was done by observing the behavior and results that application provides.

6.4.1 Creating, importing and updating timetable

These are the core three functions of the application. As it was shown in previous **Figures 5.4-5.9** of creation of timetable takes few steps. In its turn, importing timetable requires only network connection and a single click. As the result, users are able to view and always refer to this timetable. Even though, the steps are different, but the result remains the same. This means that application provides absolutely same interface for timetable editing. Timetable editing uses the data from already existing timetable in order to make changes to it.

The result of these steps all together is satisfactory, because from the functionality point of view everything work properly.

6.4.2 Adding assignment and assignment notification

Assignment data is maintained in separate table inside the database. But, it refers to the main timetable data in order to link assignment with the corresponding subject. Also, a background thread is implemented in order to send notifications to the user about upcoming assignments.

As the result, same as the previous scenario, from functionality point of view adding, viewing and sending a notification is working properly, except the fact that is a little bit difficult in controlling the background thread working more efficiently.

6.4.3 Putting altogether

In general, all parts and module of the application are interrelated in between, but without created timetable other functions do not work, because everything is based on the data of the timetable.

Application as a whole functions perfectly. There are only two functions that are needed to be maintained in the future, which are exporting timetable and working with background threads. The problem with exporting timetable that is a screenshot is made for the area that fits into the screen. So, the rest of the area remains not being exported. A problem with background threads has already been described in the section earlier.

CHAPTER 7

CONCLUSION

7.1 Project Achievements

During this project total number of six chapters is written in the form one whole thesis.

In Chapter 1, general background, objectives and scope of the project is provided.

Chapter 2 came with detailed research in different areas in order to identify and satisfy project requirements.

Chapter 3 is all about analyzing identified requirements.

Chapter 4 covers designing the user interface and come up with different types of diagrams, such as DFD, ERD and sequence.

Chapter 5 was about how the application is implemented by providing screenshots of the developed application together with detailed explanation and corresponding code snippets.

In Chapter 6 three types of testing are applied, which are unit, usability and integration testing.

Generally, during the development of this project the huge amount of knowledge and experience is earned.

7.2 Problems Encountered

- **Lack of experience in developing Android based applications:**

  In the beginning of the project the lack of knowledge in developing android application existed. This caused the time spending on learning of how to develop android based applications.

- **Limitations of the Android platform**

  Android platform has its own limitations as a mobile device, since the resources and screen size are limited. Also, application of some widgets causes few problems, like resizing checkboxes, controlling background threads and so on.

7.3 Future enhancements

- **Improve performance speed:**

  The performance speed is needed to be enhanced because there are a lot of I/O operations to the database which cause the performance to be slow. Future enhancement should be done in order to improve algorithm of the frequency of accessing database.

- **Improve UI design:**

  In order to provide good and user friendly user interface it must be improved in the future. Some activities must provide better usability in order to correspond to principles of Heuristic Evaluation.

- **Make background threads more efficient:**

  As it was mentioned before, the work of the background threads is not reliable. So, future enhancement is required in order to make threads work perfectly

REFERENCES

[1] Valacich, F.J., George, F.J., Hoffer, A.J. 2009. *Essentials of System Analysis and Design.* London: Prentice Hall.

[2] David, M. 2011. *Flash Mobile Developing Android and iOS Applications.* Oxford: Elsevier Inc.

[3] Meier, R. 2009. *Professional Android™ Application Development.*Indiana: Wiley Publishing Inc.

[4] Web Innovation in Asia. 2012. Internet: http://e27.sg/2012/05/16/singapore-hits-mobile-penetration-rates-of-148-9/smartphones/. Date of extraction: 11th July 2012

[5] Google in Enterprise. 2011. Internet: http://www.techrepublic.com/blog/google-in-the-enterprise/10-billion-android-market-apps-served-and-counting/697. Date of extraction: 11th July 2012

[6] Lamsal, P. 2007. *J2ME Architecture and related embedded techonologies.* Internet: http://www.cs.helsinki.fi/u/campa/teaching/j2me/j2me.html. Date of extraction : 12th July 2012.

[7] iOS Technology Overview.2011. Internet: http://developer.apple.com/library/ios/#documentation/Miscellaneous/Conceptual/iPhoneOSTechOverview/IPhoneOSOverview/IPhoneOSOverview.html. Date of extraction: July 12th July 2012.

[8] Goldstein, N. 2010. *iPhone Application Development for Dummies.* Indiana: John Willey and Sons Inc.

[9] Android OS Architecture. 2011. Internet: http://android-shark.ru/arhitektura-operatsionnoy-sistemyi-android/. Date of extraction: July 13th 2012.

[10] A+ Timetable application. 2012. Internet: https://play.google.com/store/apps/details?id=info.timosoft.aplustimetable. Date of extraction: July 13th 2012.

[11] UWB University Timetable application. 2012. Internet: http://ru.androidzoom.com/android_applications/tools/timetable_xbde.html. Date of extraction: July 13th 2012.

[12] Homework and Time table. 2012. Internet: http://ru.androidzoom.com/android_applications/education/homework_mlvx.html?nav=related. Date of extraction: July 13th 2012.

[13] CNETDownload. 2008. Notepad++. Internet: http://download.cnet.com/Notepad/3000-2352_4-10327521.html#ixzz1hjsDPdws. Date of extraction: July 13th 2012.

[14] What is Java and why do I need it. Internet: http://www.java.com/en/download/faq/whatis_java.xml. Date of extraction: July 14th 2012.

[15] What is SQL? 2012. Internet: http://www.sqlcourse.com/intro.html. Date of extraction: July 14th 2012.

[16] What is PHP? Internet: http://php.net/manual/en/intro-whatis.php. Date of extraction: July 14th 2012.

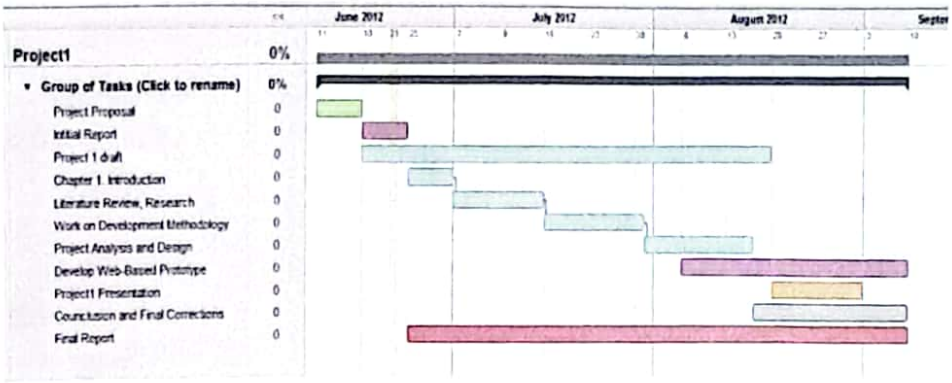
[17] What is HTML anyway? 2004. Internet: http://www.goodelIgroup.com/tutorial/chapter1.html. Date of extraction: July 14th 2012.

[18]. Marks, D.2002. *Development Methodologies Compared.* Internet: www.ncycles.com. Date of extraction: July 15th 2012.

[19] Software developing methodology. Internet: http://en.wikipedia.org/wiki/Software_development_methodology. Date of extraction: July 15th 2012.

APPENDICES

APPENDIX A: Project Schedule – Gantt Chart



APPENDIX B: List of Standard Abbreviations

app.	Application
etc.	(et cetera): and so forth
fig./figs.	figure(s)
iOS	Iphone Operating System
J2ME	Java 2 Micro Edition
Mr.	Mister