



ISSN: 0975-766X
CODEN: IJPTFI
Research Article

Available Online through

www.ijptonline.com

ANDROID PLATFORM FOR ACEDAMICS

P.Sivaram¹ and V.R.Priya²

^{1,2}Assistant Professor, School of Computing,

Department of Computer Science and Engineering, Srinivasa Ramanujan Centre, SASTRA University,
Kumbakonam, Tanjore District, Tamilnadu, India – 612001.

Email: ponsivs@src.sastra.edu

Received on: 15.10.2016

Accepted on: 12.11.2016

Abstract

Ongoing recent innovations under Android Platform, open the room for developing new applications with innovative ideas as well as migrating existing applications of another platform in to Android. The Android platform focuses on mobile computing factors and it is being an operating system for mobile phones, tablets and even for laptops. The resilience characteristics of mobile computing devices such as portability, faster operating speed, efficient productivity, ease of handling, entertainment and effective handling of business processes are encouraging to start application oriented development under android platform. In this context, educators are focusing on framing the academic courses for learning android development to enhance the students' knowledge, though which innovative ideas can be populated and based on which the social welfare applications can be created. In this work, we proposed an android app, which is useful for the students of under graduate to learn and understand the scope of laboratory course on android development and is named as Android Lab Programs.

Keywords: Android, Operating System, Mobile Computing, Programming, Android app development, Academic Courses.

1. Introduction

Teaching and learning are essential key factors for educators and students. Emerging technologies are occupying all sort of processes in any system with its own way of up gradation. The teaching-learning process in an academic system is focused in this paper, and the android platform based programming and understanding the scope of the course is given in this application. Teaching-learning process is supported by various teaching aids (may traditional and modern teaching aids), which helps the educators to deliver their contents in an effective manner to the learners; as well as it reduces the hurdles in the learners to understand and observe the contents delivered. This study aims to

propose an android app, which is based on the theory and laboratory courses named “Android Applications” and “Programming in Android”. Any laboratory course in an academic system, is framed with laboratory manual, educator’s explanation on the problems of the course contents, providing the etiquette contents on the problem and field of applications where the problem can be implemented [1]. In the traditional mode of providing training to the learners, laboratory manual and explanation from the educators assists the students to get involve in problem solving (i.e. programming). In addition to the above mentioned laboratory course training materials, this android app (Android Lab Programs) is a tool in which the learners can realize and understand,

- List of Problems in the Course (Programs to be performed in the Laboratory)
- Android based Optimized Graphical User Interface design.
- Implementing effective algorithms (Problem specific).
- Identifying and validating the input data.
- Analyzing alternative ways of approaching the problem.
- Populating new ideas from the learners point of view.
- Simplifying the steps involved in bring the solution (Problem specific).

The role of handheld internet enabled devices (smartphones, tablets and laptops) are widening their scope of usage in terms of application aspects. These devices’ portability characteristic, lead them to be used in various locations (not a fixed place of use) by ensuring availability of internet. A smartphone / tablet consists its own processor, memory and it is capable of performing manipulations like a personal computer. These devices can be used as a computational device, how the personal computer fulfills its desired needs of operation. The students’ community is impressed by smartphones and tablets and are using these devices [2]. By considering students’ community in mind, most of the applications and services in various platforms are migrated into Android to get more sophistication and customers for business [3]. Most of the business organizations are now focusing their advertisement activities with android development in the form of android app. This Android laboratory programs app will help the students to have live demonstration at their door steps when need arises. The educators can use this app as a supporting material for handling such laboratory courses [4].

Android Laboratory Course Overview

Android Platform: Android platform is fastest trending operating system (OS) with Smartphone users. In other terms, it is being a popular OS for touch screen devices such as Smartphone, tablets, specialized user interface for

televisions, android enabled vehicles, and android wear. In this scenario, plenty of opportunities are available for developing (new as well as migrating existing) various applications, which are available in other platforms. Android development is an open source platform; anyone (who know android app development) can develop applications with free of cost (also application developing tools such as IDE, Android SDK, Java, etc.). The advantages of android OS are multitasking, open source, ease of notification, easy to access to many applications via Google Play, nonspecific on OEM manufacturing vendors (not like iOS which is only for apple iPhone), modified ROM adaptation, use of widgets, and access to various essential life leading google services.

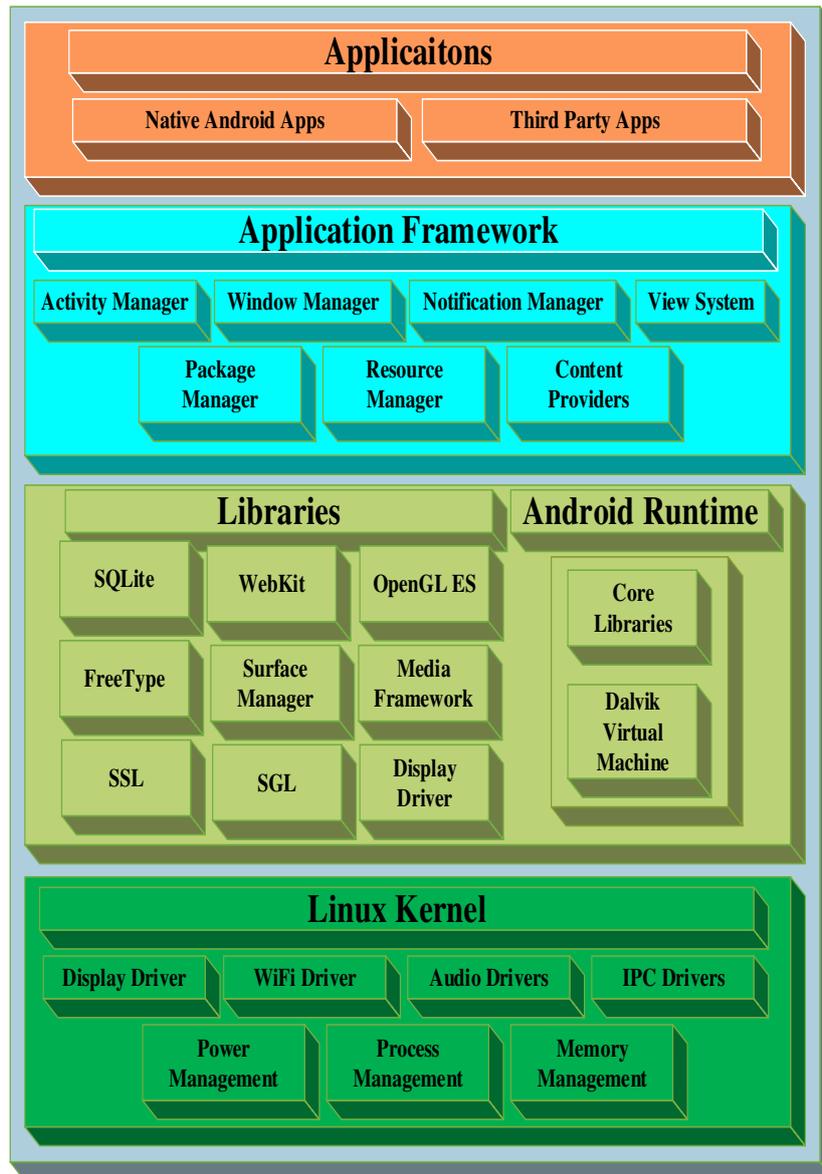


Fig. 1. Android OS architecture.

Android OS is a core and comprised of four layers [5]. The figure 1, demonstrates the Android OS architecture with four layers and their internal arrangements of components necessary for its operation. The four layers of Android OS architecture core are given in the below table.

Table 1. Android OS architecture layers.

Layer	Name of the Layer
1	Linux Kernel
2	Libraries & Android Runtime
3	Application Framework
4	Applications (Native & 3rd Party Apps)

2.2. Linux Kernel

Any Operating system is identified by its kernel. Android OS is developed with Linux kernel (Long term support (LTS) branches). Linux kernel is based on the hardware requirements of the android device. There are various kernel versions of android are existing and it is starting from the version 2.6.25. the below table.

Table 2. Linux kernels and its Android version names [6] [7].

Android Version	Android version name	API Level	Linux kernel
1.0	Apple pie	1	2.6.25
1.1	Banana bread	2	2.6.25
1.5	Cupcake	3	2.6.27
1.6	Donut	4	2.6.29
2.0	Eclair	5	2.6.29
2.0.1	Eclair	6	2.6.29
2.1	Eclair	7	2.6.29
2.2	FroYo	8	2.6.32
2.3	Gingerbread	9	2.6.35
(2.3.3) - (2.3.7)	Gingerbread	10	2.6.35
3	Honeycomb	11	2.6.36
3.1	Honeycomb	12	2.6.36
3.2, 3.2.1, 3.2.2, 3.2.4, 3.2.6	Honeycomb	13	2.6.36
4.0, 4.0.1, 4.0.2	Ice Cream Sandwich	14	3.0.1
4.0.3, 4.0.4	Ice Cream Sandwich	15	3.0.1
4.1, 4.1.1, 4.1.2	Jelly Bean	16	3.0.31

4.2, 4.2.1, 4.2.2	Jelly Bean	17	3.4.0
4.3	Jelly Bean	18	3.4.0
4.4, (4.4.1) – (4.4.4)	KitKat	19	3.10
4.4	KitKat Wearable Extensions	20	3.10
5.0, 5.0.1, 5.0.2	Lollipop	21	3.16.1
5.1, 5.1.1	Lollipop	22	3.16.1
6.0, 6.0.1	Marshmallow	23	3.18.10

2.3 Libraries & Android Runtime

In Android OS architecture, the next layer consists two major groups of components. First and second group of library components are named as android (native) libraries and android runtime. The android native library files are written with C and C++ languages and provides necessary support for the developer to reduce his burden on development work. Built-in library files are more sophisticated for a developer and handled effectively. These library files play role of an application layer activity and provides much comfort to the developer as well as the user of the system. Android native libraries include, SQLite, WebKit, OpenGL ES, Surface manager, Media framework, SSL, SGL, Display driver, etc.

The SQLite is list of library files, which are utilized to store the data in an internal database of the android system and is treated as database engine. The Web Kit library files are used to display html contents on your android system and is treated as browser engine. The OpenGL ES library files are used by android system to display and render 2D and 3D graphics. The Surface manager is performing a task called off screen buffering and helps the system to display the windows in a unique manner in the appearance. The media framework consists various codes of audio and video formats, which are helpful to playback and record audio/video with the help of the android system [8]. Until, the lollipop version’s release, android OS has Dalvik Virtual machine (DVM) and java core libraries in separated mode. DVM is a virtual machine, like JVM (Java Virtual Machine), which is used in android systems to execute programs with low resources utilization and improved energy savings. It allows multiple copies of virtual machines to run at the same time with all common functionalities are included. Java core libraries are used in the construction of manipulation part (problem solving steps (in the program code)) of the program [8].

2.4 Application Framework

The third layer in this architecture is the application framework. All application related activities are interacting with the system with the help of this layer. The application framework consists activity manager, window manager, notification manager, view system, package manager, resource manager, content provider, etc. This layer is responsible for all types of system and third party app related activities with the system. For example, this layer takes care of the following:

- Data sharing between two different applications in a same system.
- Voice call management.
- Location Management.
- Resource Management.

2.4.1 Applications

In an Android system applications are called as apps and there are two types of apps. The apps which are managing OS related process are System apps and in general they are called native android apps. Other than the system managing apps, the apps installed in the system from google play or android market are known as third party apps. These apps are having part in the Android OS over the Application framework and is treated as applications layer of this architecture.

3. Android Laboratory Programs App Architecture

Instead of traditional mode (with laboratory manuals) of teaching the laboratory courses in under graduate degree program, an Android app is used to demonstrate the list of problems and way to approach to reach their solutions [9]. A clear view of the logical correctness of problem fix and graphical user interface can be instructed to the students with the help of it (application). This work is focusing on Android based laboratory programs demonstration. The list of experiments in the course is restricted to twelve in numbers. They are listed in table 3.

Table 3: List of Android Lab Experiments.

Sl.No.	Name of the Experiment
1	Perform Arithmetic Operations with two variables.
2	Perform Swapping of two numbers with two variables.
3	Identify and display, whether given number is odd or even number.
4	Find the biggest and smallest number from the given three numbers.

5	Find a student's any one subjects' university grade value with necessary marks as input.
6	Find the factorial value of a given number.
7	Find the summation value of a given number.
8	Find the Fibonacci series of a given number.
9	Find and list all odd numbers between the given number range.
10	Find and list all even numbers between the given number range.
11	Create an app to demonstrate Login and Registration operations.
12	Create an app to demonstrate online database transactions of information.

Android Lab Programs (ALP) is the name of the application and is developed using android platform. Android development has many tools, and are Integrated Development Environment (IDE), Android Software Development Kit (Android SDK), and Java Development Kit (JDK). There are many IDEs' available in the market among them one of the best and open source IDE is Android Studio, which is provided by developer.android.com [10]. In this web link, the combined bundle of IDE, Android SDK, (i.e. Android Studio with Android SDK) is available for download as well as separate IDE and Android SDK [11]. The JDK is an open source tool which can be downloaded from oracle.com [12]. Android Studio is an intelligent code editor, in which much of the corrections in the program code are taken care and alternate suggestions to the developer are provided by the IDE itself. The project folder consists various files among with it and important files are AndroidManifest.xml, MainActivity.java, and activity_main.xml. Training materials and tutorials for the basics of android development is available in developer.android.com [13].

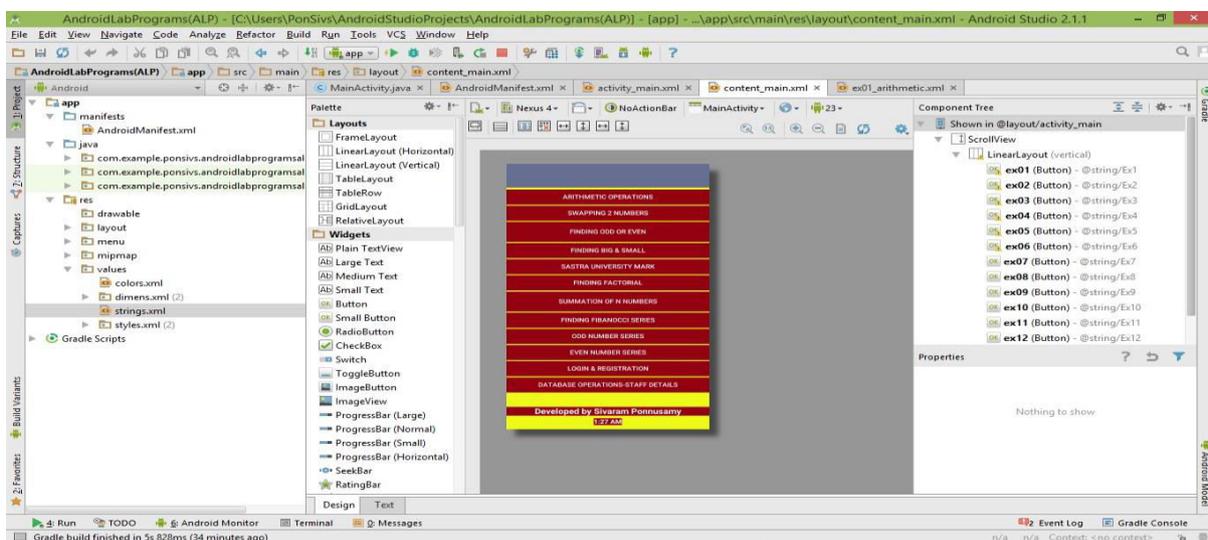


Figure 2: Android Studio Integrated Development Environment Screen (IDE).

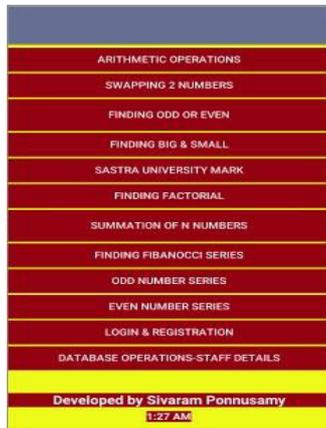


Fig. 3. (a) Home Screen Development (IDE).



Fig. 3. (b) Execution on the Emulator (AVD).

In the development of home screen of this app, 12 numbers of Button widgets are used to fill the page with enhanced user interface. Each button widget carries individual experiment screen display and the user can further manipulate appropriate experiments execution. The Indents from android development are used to create the home screen manipulation.

2.5 App Development

The first experiment in the ALP is arithmetic operations, which include addition, subtraction, multiplication, division and modulo-division. The arithmetic functionalities of the device processor are utilized in this experiment execution. The basic logic behind this experiment are known to students and only unknown is how to do with android development. A GUI is prepared with android studio xml file, which resides in the project folder named as \AndroidLabPrograms(ALP)\app\src\main\res\layout\ex01_arithmetic.xml. The page is set into linear layout mode and the widgets named EditText, TextView, Buttons are used to enhance the user interaction with the app. In total, 2 numbers of EditText widgets, 2 numbers of Button widgets, 10 numbers of TextView widgets are used in the app and provided with application specific labels (refer Fig.4(a)).



Fig. 4. (a) Development Screen (IDE).



Fig. 4. (b) Arithmetic Operations Execution (AVD).

As for the manipulation part, a java file is used inside the project folder and is named as \ Android Lab Programs(ALP)\app\src\main\java\com\example\ponsivs\androidlabprogramsalp\Ex01_Arithmetic.java. Two floating point variables are considered under java programming language and treated as input values for the processing the experiment. From the user interface xml file, the widgets are assigned with unique ID and are used in the java file to identify the values for further manipulation. The buttons are labeled with Calculate and Clear based on context specific. The manipulation logic for the Calculate button is collect two input float variables and perform addition, subtraction, multiplication, division, and modulo-division based on the algorithms and display results in the appropriate Text View widgets (assigned with unique ID) (refer Fig.4(b)). Except button widget, all are string data type based widgets and type conversion is essential in the manipulation logic. The Clear button is used to assign null string to the widget values for subsequent processing. The second experiment is titled with swapping of two numbers and the basic logic behind is just swapping the values of two variables each other. For this experiment user interface is a page in the form of xml file (ex02_swapping.xml), which resides in the project folder. The page is set into linear layout mode and the widgets named EditText, TextView, Buttons are used to enhance the user interaction with the app. In total, 2 numbers of EditText widgets, 2 numbers of Button widgets, 3numbers of TextView widgets are used in the app and provided with application specific labels (refer Fig.5(a)).

In the manipulation part, a Ex02_Swapping.java file is used to swap the numbers. Two floating point variables are considered under java programming language and treated as input values for the processing the experiment. The buttons are labeled with Swapping and Clear based on application specific. The manipulation logic for the Calculate button is collect two input float variables and perform swapping of two numbers based on the simple swap algorithm and display results in the appropriate TextView widgets (assigned with unique ID) (refer Fig.5(b)). Except button widget, all are string datatype based widgets and type conversion is essential in the manipulation logic. The Clear button is used to assign null string to the widget values for subsequent processing.



Fig. 5. (a) Development Screen (IDE).



Fig. 5. (b) Swapping Execution (AVD).

The third experiment named as odd or even number which explains the odd number that cannot be divisible by 2 and also ends with 1,3,5 or 9. Similarly even number is the one which is divisible by 2 and those numbers end with 0,2,4,6 or 8. For this experiment the page in the form of xml file is stored in project folder and the page is fixed into the linear layout and the elements EditText, TextView and Button are embedded into form to improve the user interaction with this application. In the app, totally 1 Edit text, 2 Buttons, and 2 no of TextView(s) are placed with specific label refer fig6 (a).

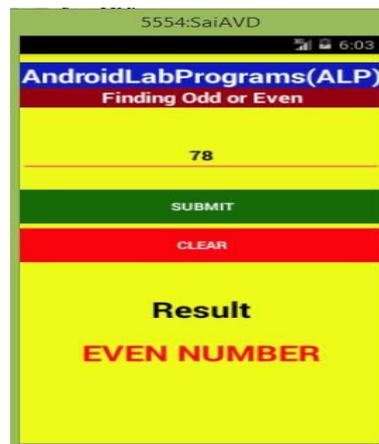
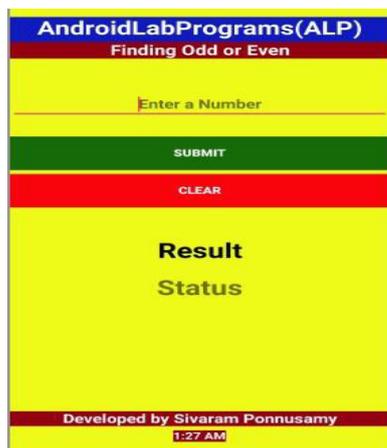


Fig. 6. (a) Development Screen (IDE).

Fig. 6. (b) Odd or Even no Execution (AVD).

In the manipulation part, java file is used inside the project folder and is named as \AndroidLabPrograms(ALP)\app\src\main\java\com\example\ponsivs\androidlabprogramsalp\Ex03_OddorEven.java. Integer variable are consider as input value in the application. The buttons are labeled as Submit, Clear based on application specific. The manipulation logic for this experiment is input an integer value in EditText and display the Result in respective TextView by pressing Submit button (assigned with unique ID) (refer fig 6.b). Except the button control, all are string datatype based widgets and type conversion is essential in the manipulation logic. The Clear button is used to assign null string to the component values for subsequent processing.

The Fourth Experiment entitled with Finding Big and Small number which exhibit the biggest number and smallest number among three numbers. The routine of finding superior value and lower value of subset of multi labels are used in this experiments [14]. Students get trained to perform in the good feature subset in future. The basic concept behind the experiment is to find the Minimum number and Maximum number among three numbers. A GUI is prepared with android studio xml file, which resides in the project folder named as \AndroidLabPrograms(ALP)\app\src\main\res\layout\ex04_bigsmall.xml. The page is set into linear layout mode and the widgets named EditText, TextView, Buttons are used to beautify the user interaction with the application. In total,

3 numbers of EditText widgets, 2 numbers of Button widgets, 2 numbers of TextView widgets are used in the app

and provided with application specific labels (refer Fig.7(a)).



Fig. 7. (a) Development Screen (IDE).

Fig. 7. (b) Big or Small Number Execution (AVD).

In the Manipulation part, a java file is used inside the project folder and is named as `\AndroidLabPrograms(ALP)\app\src\main\java\com\example\ponsivs\androidlabprogramsalp\Ex04_BigSmall.java`.

Three Integer variables are considered under java programming language and treated as input values for the processing the experiment. From the user interface xml file, the widgets are assigned with unique ID and are used in the java file to identify the values for further manipulation. The buttons are labeled with Submit and Clear based on context specific. The manipulation logic for the Calculate button is to gather all three input integer variables and execute max-min process based on the algorithms and display results in the appropriate TextView widgets (assigned with unique ID) (refer Fig.7(b)). Except button widget, all are string datatype based widgets and type conversion is essential in the manipulation logic. The Clear button is used to assign null string to the widget values for subsequent processing.

The Fifth Experiment illustrate the application development of university student mark list. This application exhibit how a student's marks can be calculated in their end semester and facilitate automatic, efficient process of result [15]. The perception of the mark calculator is exactly where the students stand in their course. The GUI is prepared with android studio xml file, which resides in the project folder named as `\AndroidLabPrograms(ALP)\app\src\main\res\layout\ex05_univmarkcalc.xml`. The page is set into linear layout mode and the widgets named EditText, TextView, Buttons are used to highlight the user interaction with the application. In total, 5 numbers of EditText widgets, 2 numbers of Button widgets, 8 numbers of TextView widgets are used in the app and provided with application specific labels (refer Fig.8(a)).



Fig. 8. (a) Development Screen (IDE).



Fig. 8. (b) University mark calculation (AVD).

In the Manipulation part, a java file is used inside the project folder and is named as `\AndroidLabPrograms(ALP)\app\src\main\java\com\example\ponsivs\androidlabprogramsalp\Ex05_UnivMarklistCalc.java`. Five Integer variables are considered under java programming language and treated as input values for the processing this application. From the user interface xml file, the widgets are assigned with unique ID and are used in the java file to identify the values for further manipulation. The buttons are labeled with Submit and Clear based on context specific. The manipulation logic for the submit button is to calculate all five input integer variables and execute mark process, Grade obtained, Result of subject based on the algorithms and display results in the appropriate TextView widgets (assigned with unique ID) (refer Fig.8(b)).

The student’s measurement obtained by grade which is determined by CGPA that is calculated by dividing the cumulative sum of grade point by summation of credit point of courses. Except button widget, all are string data type based widgets and type conversion is essential in the manipulation logic. The Clear button is used to clear the widget values for subsequent processing.

The Sixth experiment is factorial discovering strategy where app developed for factorial of number. The principal behind the program is how a number gets factorized. Students are trained to get the grouping of n digits that will be changed over to a stage of n straightforwardly [16]. Those things mainly used in the field of mathematics especially in permutations, distinguishable permutations, combinations, binomial theorem, etc along in the computer field extends as polygon shading, polynomial placement and also in designing 3d graphics for 3d games. The user interface is prepared with android studio xml file, which resides in the project folder named as `\AndroidLabPrograms(ALP)\app\src\main\res\layout\ex07_facotiral.xml`. The page is set into linear layout mode and the widgets named EditText, TextView, Buttons are used to elevate the user interaction with the application. In total,

1 numbers of EditText widgets, 2 numbers of Button widgets, 1 numbers of TextView widgets are used in the app and provided with application specific labels ().

In the Manipulation part, a java file is used inside the project folder and is named as \AndroidLabPrograms(ALP)\app\src\main\java\com\example\ponsivs\androidlabprogramsalp\Ex05_factorial.java. A single variable is considered under java regarded as input values for handle the analysis. From the user interface xml file, the objects are allotted with distinct ID and are utilized as a part of java document to distinguish the values for the advance execution. The buttons are defined with Calculate and Clear based on context specific. The manipulation logic for the Calculate button is handled the input integer variables and perform factorial based on the algorithms and display results in the appropriate TextView widgets (assigned with unique ID) (refer Fig.9(b)). The Clear button is used to assign zero to the widget values for later processing. The seventh module explore the summation of n numbers. The concept behind the programme is the infinite series whose terms are the natural number $1+2+3+4+5+n$ is a divergent series. Instead of using looping statement in the programme the recursion is applied to generate n number of series. The user interface is prepared with android studio xml file, and the page is set into linear layout mode and the widgets named EditText, TextView, Buttons are used to improve the user interaction with the application. In total, 1 numbers of EditText widgets, 2 numbers of Button widgets, 1 numbers of TextView widgets are used in the app and provided with application specific labels. In the Manipulation part, a java file is used in the project folder and is labeled as \Android LabPrograms (ALP)\app\src\main\java\com \example\ponsivs\ androidlab programs alp \Ex11_sumofnnumber. A single variable is considered under java treated as input values for handle the operation that produces the summation of n values. From the user interface xml file, the components are placed with distinct ID and are used as a part of java document to distinguish the values for the next execution. The buttons are defined with Calculate and Clear based on context specific. The manipulation logic for the Calculate button is handled the input integer variables and perform summation based recursive algorithms and display results in the appropriate TextView widgets (assigned with unique ID). The Clear button is used to clear the widget values for next operation. The eight module describes the Fibonacci sequence model which are nature's numbering framework that can show the entire structure of nature such that florets of blossoms to the bracts of pinecone. The idea behind the programme is the successive number found by the summation of two numbers before it and first number, second number will be filled and successive number will be printer. The sequence of number starting from either 0 or 1 that listed as 0, 1, 1, 2, 3, 5, 8, 13, 21, 34...[17]. and the Rule is $x_n = x_{n-1} + x_{n-2}$.

The user interface is prepared with android studio xml file, which belongs in the project folder named as \AndroidLabPrograms(ALP)\app\src\main\res\layout\ex07_fibonacci.xml. The page is set into linear layout mode and the widgets named EditText, TextView, Buttons are used to elevate the user interaction with the application. In total, 1 numbers of EditText widgets, 2 numbers of Button widgets, 1 numbers of TextView widgets are used in the app and provided with application specific labels

In the Manipulation part, a java file is used in the project folder and is labeled as \AndroidLabPrograms(ALP)\app\src\main\java\com\example\ponsivs\androidlabprogramsalp\Ex10_fibonnacii.java.

A single variable is considered under java treated as input values for handle the operation. From the user interface xml file, the components are placed with distinct ID and are used as a part of java document to distinguish the values for the next execution. The buttons are defined with Calculate and Clear based on context specific. The manipulation logic for the Calculate button is handled the input integer variables and perform factorial based on the algorithms and display results in the appropriate TextView widgets (assigned with unique ID) (refer Fig.10(b)). The Clear button is used to clear the widget values for next operation.

The ninth and tenth module illustrate the generation of odd and even numbers and those are only suitable for whole numbers. Some facts about the odd and even listed below

For addition: Even+Even=Even;

 Even+Odd=Odd;

 Odd+Odd=Odd

For subtraction: Even – Even=Even;

 Even – Odd=Odd;

 Odd – Odd=Even

For multiplication: Even X Even=Even;

 Even XOdd=Even;

 Odd XOdd=Odd

But when those numbers are divided it produce some peculiar problems and it should be eliminated for further execution. The user interface is prepared with android studio xml file, and the page is set into linear layout mode and the web elements are namedwith EditText, TextView, Buttons are used to improve the user interaction with this application. In total, 1 number of EditText widgets, 2 numbers of Button widgets, 1 number of TextView widgets are

used in the app and provided with application specific labels. In the Manipulation part, a java file is used in which a single variable is considered under java treated as input values for handle the operation that produces the series of odd and even numbers. A special function used to display the odd and even series in the respective TextView subsequently. The buttons are defined with Calculate and clear based on context specific. The manipulation logic for the Calculate button is handled the input integer variables and perform summation based recursive algorithms and display results in the appropriate TextView widgets (assigned with unique ID). The eleventh Section explore the procedure of entering login to the registration form which is eye opening to the students obtaining access to the computer by individual and grant authentication themselves. The user name and password is the user credentials used to register a form. The android studio xml file and the page is set into linear layout mode and the web elements are named with Edit Text Buttons are used to enhance the user interaction with this application. Totally 2 numbers of Edit Text, 2 no of Buttons utilized in the application development. In the manipulation section a java file named is register.java in which two values considered as input for evaluation of register form. In the first button named as Submit which evaluate the username, password from Edit Text and the next named Clear utilized to clear the text for next successive execution. The last section explains the database management system that educate students about manipulation of database activities, relation between application and database [18]. This module clearly discloses basic work and necessity of database in the company hold large set of data.

4. Conclusion

This paper strives to efficiently provide and derive the explanation for the students' lab exercises. It aims at initiating the idea of factorial and fibonacci problems along with the 'odd or even' problem which offers the students' the literal idea behind solving much difficult problems in mathematics. Further it lends a hand for the students' by explaining the computation of University scores. As an added note, this paper also uses certain DBMS concepts such as insertion, deletion and also data manipulation.

References

1. Se Jun Song, "Lessons learned from Mobile Computing Application Development with Android", (AC 2012-3701), American Society for Engineering Education, 2012.
2. Suliman S. Aljomaa, Mohammad F. Al.Qudah, Ismael S. Albursan, Salaheldin F. Bakhiet, Adel S. Abduljabbar, "Smartphone addiction among university students in the light of some variables", Computers in Human Behavior, Volume 61, pp.155-164, August 2016.

3. Christopher Helf and Helmut Hlavacs, "Apps for life change: Critical review and solution directions", *Entertainment Computing*, Volume 14, pp.17-22, May 2016.
4. Hind M. Al-Otaibi, Reem A. AlAmer, Hend S. Al-Khalifa, "The next generation of language labs: Can mobiles help? A case study", *Computers in Human Behavior*, Volume 59, pp.342-349, June 2016.
5. Dimitris Geneiatakis, Igor Nai Fovino, Ioannis Kounelis, Paquale Stirparo, "A Permission verification approach for android mobile applications", *Computers & Security*, Volume 49, pp.192-205, March 2015.
6. James Steele, Nelson To, Shane Conder, "The Android Developer's Collection (Collection)", Addison-Wesley Professional, 2011.
7. "Android version history", https://en.wikipedia.org/wiki/Android_version_history, 2016.
8. "The Android Source Code", <http://source.android.com/source/index.html>, 2016
9. Mohamed Sarrab, Mahmoud Elbasir, Saleh Alnaeli, "Towards a quality model of technical aspects for mobile learning services: An empirical investigation", *Computers in Human Behavior*, Volume 55, Part A, pp.100-112, February 2016.
10. "Android Developers", <https://developer.android.com/index.html>, 2016.
11. "Android Studio", <https://developer.android.com/studio/index.html>, 2016.
12. "Java SE Downloads", <http://www.oracle.com/technetwork/java/javase/downloads/index.html>, 2016.
13. "Meet Android", <https://developer.android.com/studio/intro/index.html>, 2016.
14. Yaojin Lin, Qinghua Hu, Jinghua Liu, and Jie Duan, "Multi-label feature selection based on max-dependency and min-redundancy", *Neurocomputing*, Volume 168, pp.92-103, November 2015.
15. Eyo O. Ukem, Francis A. Ofoegbu, "A Software application for University Students Results Processing", *Journal of Theoretical and Applied Information Technology*, Vol. 35 No.1, pp.34-43, January 2012.
16. A.M. Ibrahim, "Extension of factorial concept to negative numbers", *Notes on Number Theory and Discrete Mathematics*, Vol. 19, pp.30-42, 2013.
17. Morteza Esmaeili, "More on the Fibonacci sequence and Heisenberg matrices", *Electronic journal of combinatorial number theory* 6, pp.1-8, 2006.
18. Yvette E. Gelogo and Sunguk Lee, "Database Management System as a Cloud Service", *International Journal of Future Generation Communication and Networking*, Vol. 5, No. 2, June 2012.