**Final Report - Fun Box**

Team Name : **STARS**

Team Members : SATYA VENKATA KRISHNA ACHANTA

ROBIN KUMAR

SHIVA SAI GOUD PABBA

Title : FUN BOX

Category : ANDROID BASED MOBILE APPLICATION

Key words : Monitoring each screen

Game play

Menu

**Teams Members Contribution in SDLC order:**

**Robin Kumar: almost 30% of project work done by him**

Documentation of project requirement and analysis

Designing of the Application using Android Studio

**Satya Venkata Krishna Achanta:40-45% of contribution**

Implementation of project with detail execution process

Completing Document for testing guidelines

Final Documentation

**Shiva Sai Goud Pabba:30-35% of contribution**

Testing and Conclusion

Documentation of User Manual

**ABSTRACT**

Smart phones have opened mobile gaming to a wide audience, beyond the hard-core gamer. In this class, we focus on casual games that each that broader audience. The class begins with an introduction to building Android applications with Google's SDK and Eclipse. Students discover the basics of creating layouts and custom views, playing music and sound effects, and getting user input from the touch screen and accelerometer. We also discuss how simple mechanics can make for immensely satisfying games. There is class discussion and readings on the success of casual games such as Bejewelled, Tetris, and Snood. Intro to Computational Media or equivalent programming experience is required.

Fun box is also a mobile game application which is supported by the android devices.

It is a packed application that itself consists small number of guess games.

* It will have different functionality for user.
* When user enters into application Fun Box will have three different Games

Such as Monty hall game, other small application.

* Number guessing game: it gives a lucky number with quote by taking some information from user.
* Monty hall game : it is a puzzle game that user have to choose a door that has a prize behind it Among other two doors.
* Buddy search : it give you best friend name according to given set inputs.

**1. INTRODUCTION**

The main motto of this project is to explore the concepts of mobile application development in android. Game applications are the most liked and downloaded applications from android market. A game application that is simple and easy to be used in a mobile handset can become a hit with millions of mobile users.

**1.1 Overview**

The motive of this application is to learn and experience developing a simple game application in android targeting a small set of users. The knowledge obtained in this process can be applied later in the career to develop any similar kind of application focusing large group of users.

**1.2 Goal:**

The project is to develop an android game application targeting children as the audience. Android is one of the leading and fastest growing mobile platforms today. Android is an open source software and the Android SDK is available to developers for free. The minimal setup time and the number of tools available for android SDK to ease the development process allows the developers to concentrate in the design and implementation details of the application in the available timeframe. The main objective of this project is to experience developing an android mobile application which is one of the booming trends in computing industry. The application helps to improve the concentration of the player.

The goal of this project is to develop a simple game application for kids on an android based mobile phone. In this application user can find the below 3 options

* When user enters into application Fun Box will have three different Games

Such as Monty hall game, other small application.

* Number guessing game: it gives a lucky number with quote by taking some information from user.
* Monty hall game : it is a puzzle game that user have to choose a door that has a prize behind it Among other two doors.
* Buddy search : it give you best friend name according to given set inputs.

**1.3 System Requirement Specification**

Software Requirement Specification (SRS) is the starting point of the software developing activity. As system grew more complex it became evident that the goal of the entire system cannot be easily comprehended. Hence the need for the requirement phase arose. The software project is initiated by the client needs. The SRS is the means of translating the ideas of the minds of clients (the input) into a formal document (the output of the requirement phase.)

**1.4 Data collection:**

**Observation of the Existing System:**

We do have so many individual gamming applications in the current world. With the help of this application user can experience 3 games under a single application called Fun Box. By taking all those games as reference we developed this project with additional features such as start menu, sound effects, scores and multiple apps.

**1.5 The SRS phase consists of two basic activities:**

**Problem/Requirement Analysis:**

The process is order and more nebulous of the two, deals with understand the problem, the goal and constraints.

**Requirement Specification:**

Here, the focus is on specifying what has been found giving analysis such as representation, specification languages and tools, and checking the specifications are addressed during this activity.

The Requirement phase terminates with the production of the validate SRS document. Producing the SRS document is the basic goal of this phase.

**1.6 Purpose & Scope**

**Purpose:**

The main objective of this project is to experience developing an android mobile application which is one of the booming trends in computing industry. The application helps to improve the concentration of the player. It’s very curious to know about your best friend name or your lucky number by using your date of birth. By taking that advantage this application developed.

**Scope:**

Android is one of the leading and fastest growing mobile platforms today. Android is an open source software and the Android SDK is available to developers for free. The minimal setup time and the number of tools available for android SDK to ease the development process allows the developers to concentrate in the design and implementation details of the application in the available timeframe. This gaming application developed so that it can easily run on any android based smartphones.

**1.7 Goals & Objectives**

1. It has interactive start screen with player.
2. Screens should be maintained to Players in an efficient manner.
3. To illustrate each and every object with specific action and look and feel mode to the Player.
4. Allow user to select favorite application with the help of buttons
5. In Buddy Search application, accepts only name(First name) in text field, otherwise show error message
6. In Monty Hall application, displays game rules & allow users to select a door among three
7. Lucky number, allows user to enter his day of date of birth
8. Display all results once taken valid input
9. Ability to come back to main menu by anytime

**2 Literature**

****

**JAVA**

Java was conceived by James Gosling, Patrick Naughton, Chris Warth, Ed Frank and Mike Sheridan at Sun Microsystems Inc. in 1991. It took 18 months to develop the first working version. This language was initially called “Oak” but was renamed as “Java” in 1995. Between the initial implementation of Oak in the fall of 1992 and the public announcement of Java in the spring of 1995, many more people contributed to the design and evolution of the language.

Java is an object-oriented programming language with a built-in application programming interface (API) that can handle graphics and user Interfaces and that can be used to create applications or applets. Because of its rich set of API's, similar to Macintosh and Windows, and its platform independence, Java can also be thought of as a platform in itself.

In Java we distinguish between applications, which are programs that perform the same functions as those written in other programming languages, and applets, which are programs that can be embedded in a Web page and accessed over the Internet. Our initial focus will be on writing applications. When a program is compiled, a byte code is produced that can be read and executed by any platform that can run Java

The main properties of the java, which made java so popular, are as follows:

1. Simple
2. Secure
3. Portable
4. Object-oriented
5. Robust
6. Multithreaded
7. Interpreted
8. High Performance
9. Distributed
10. Dynamic

.Java has standard libraries for solving specific tasks. These tasks include

- Networking

- Database Connection (via JDBC)

- Multithreading

-Android

**2.1 Java’s Magic: the Byte Code**

The key that allows Java to solve both the security and the portability problems just described is that the output of a Java compiler is not executable code. Rather, it is Byte code. Byte code is a highly optimized set of instructions designed to be executed by the Java run-time system, which is called the Java Virtual Machine (JVM). That is, in its standard form, the JVM is an interpreter for Byte code. This may come as a bit of a surprise.

Translating a Java program into Byte code helps makes it much easier to run a program in a wide variety of environments. The reason is straightforward: only the JVM needs to be implemented for each platform. Once the run-time package exists for a given system, any Java Program can run on it. Remember, although the details of the JVM will differ from platform to platform, all interpret the same Java Byte code.

**2.2 ANDROID:**

The project needed several requirements to be gathered before proceeding to design the game application. One among them includes gathering information about the suitable mathematical order that would be easy for a normal user and easy to understand the goal of the game. To identify this I searched the existing android applications in the market and gathered information about what kind of application has been done so far and what not has been attempted.

It was also noted that more than 90% of android device users have android operating system of version 2.1 or later. The functionality requirements of the application are simple and straight forward whereas the technical details involved some research on the available android technologies.

GUI is critical for any game application and Android has multiple facilities for drawing GUI for games. Few study materials in android to code with the concepts like surface view and canvas were also gathered so as to smooth the application development process. These gathered information helped to provide a clear direction of deciding the software requirements and hardware requirements.

To check the feasibility of using these technologies in android for developing the proposed game, I practiced developing few simple android applications using these concepts. One of them include an application to Calculate lucky number Game by taking user input as series of numbers, The gathered materials and the exercise problems I attempted prepared me to start designing the actual application called ‘Android game using random action called FUN BOX’.

So by using java it is possibility to design any kind of android application. With this we used eclipse as environmental main aspect with this we need android SDK and emulator.

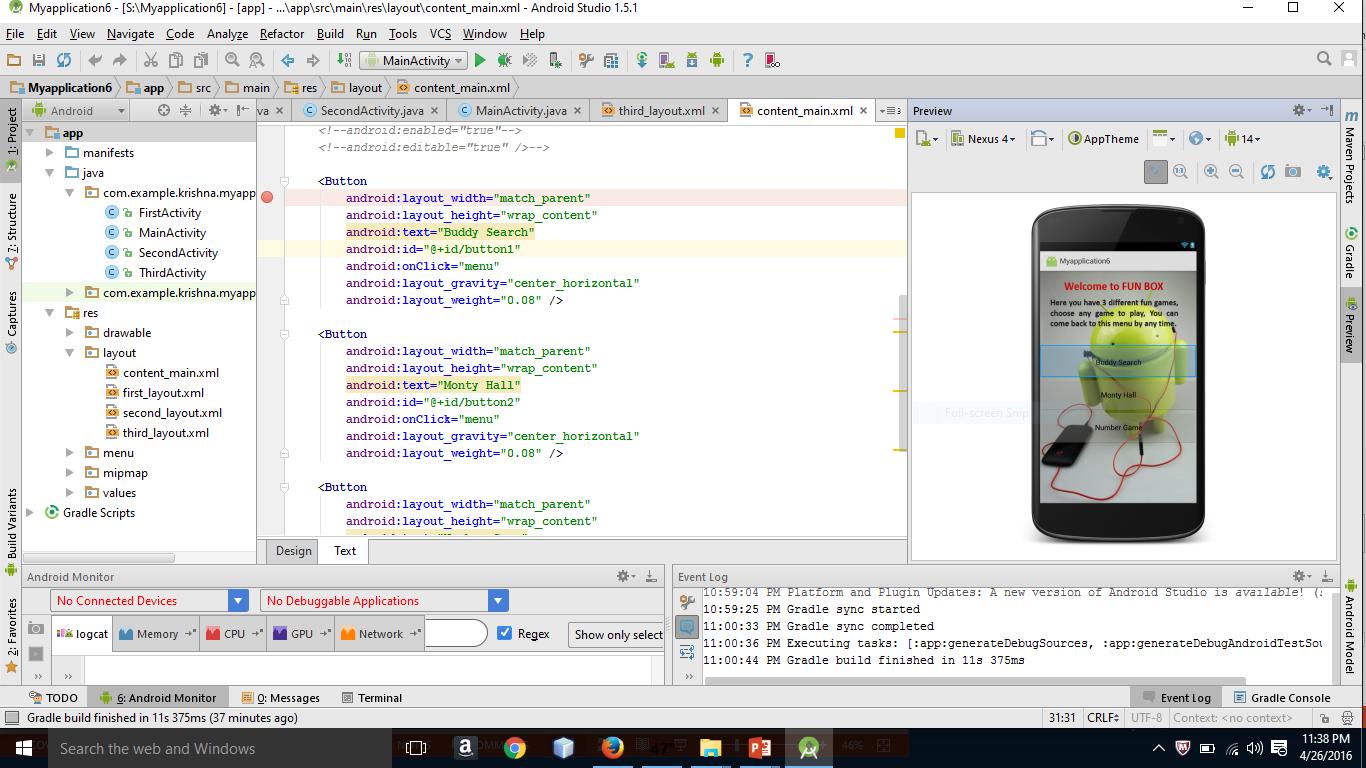
It is sure that android code comparatively different from java code, but it takes all the motivation from java with addition to this it has additional libraries and it has so interactive features by using layouts and xml documents.

**2.3 XML**

XML is a king of markup language called Extensible Markup Language. By using this we can design how a user interface looks like with layouts. It takes small amount of percentage in this code.

Basically xml have some précised tags and user tags also, by using these tags a user have to determine what king of expressions he/she need. And it is the part of HTML (Hypertext Markup Language)

An example of a xml script in our application is shown below



**3 ANALYSIS & DESIGN**

Analysis is the detailed study of the various operations performed by a system and their relationships within and outside of the system.

During analysis, data are collected on the available files, decision points, and transactions handled by the present system.

This section attempts to describe each Activity of the project in brief, and the detailed description of each of these Activities and Actions is spread throughout this document.

This Squeeze Box Gaming Application consists different Sectors, They are

* Start Screen

It have 3 utility options as described below, user can choose what application he want to play from FUNBOX app

* Number guessing game
  1. Taking Input

In order to produce a result it takes required input from the user

* 1. Displays number

If user gives right answer or exceeds max number of tries it appears with appropriate results

* Buddy search
  1. Asking for name of user

It process the input with some certain logic

* 1. Prints name

It displays the result to the user and first letter of your friend

* Monty hall
  1. Picks a door number

User can choose any door from total number of doors

* 1. Confirmation

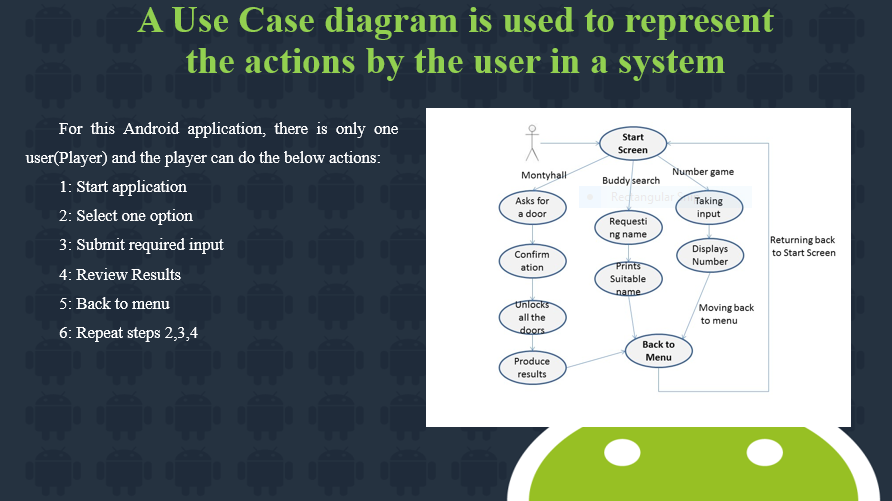
Asks for final decision to switch the door with other

* 1. Unlocks all doors

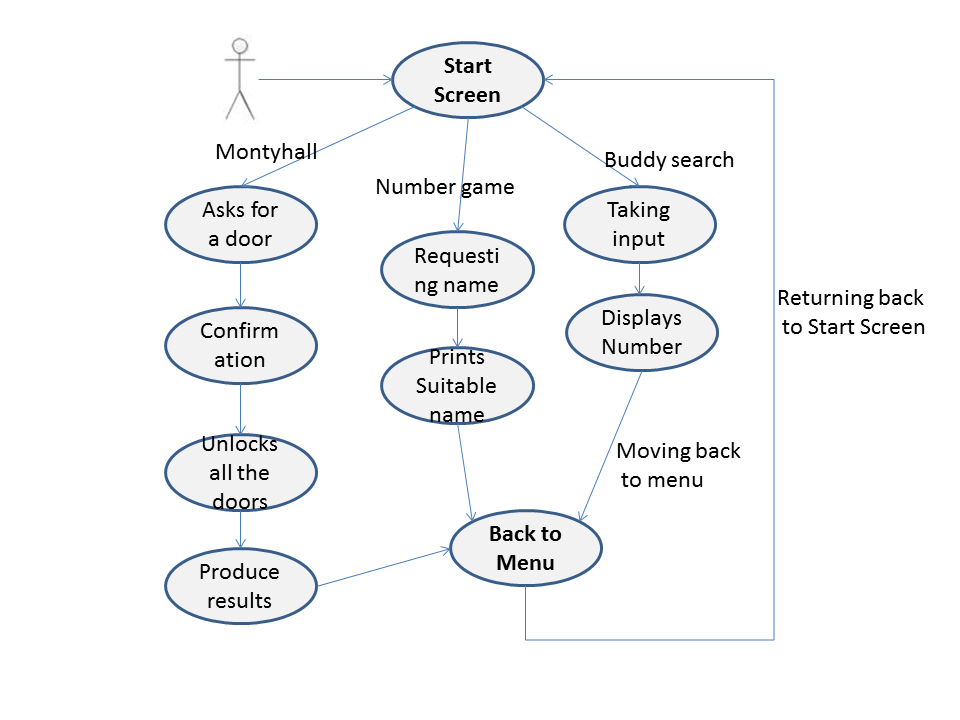
It displays all doors

* 1. Produces result

It produces prizes based on selected results

**3.1 Coming to design, **

User have the above actions, he can start the application by simply clicking application icon followed by all other actions like select a game, enter inputs and other.



**3.3 Advantages of Using UML:**

As the name suggests UNIFIED MODELING LANGUAGE. Modeling has been around for years, not only in software field but also in other trades like civil, mechanical etc. example in civil engineering drawing the main architecture built of diagram is a model by itself. Modeling makes a complex and huge system to break up in to simple and discrete pieces that can be individually understood. Example simple flow drawing is modeling.

There are two main advantages of modeling:

* Readability: Representing your whole architecture in flowchart, class diagrams, ER diagrams etc makes your project more readable. Especially when programmers change jobs handover becomes easier. More the project is not readable more the dependencies.
* Reusability: After the system is more readable and broken down into pieces, it becomes easier to identify redundant and similar modules. Thus increasing reusability.

**3.4 Components of Android SDK:**

SDK means Software Development Kit, and here there exist different kinds of components like.[R3].

Android virtual device

Android Layout manager

Xml coding

Android Manifest.xml

Activities

Services

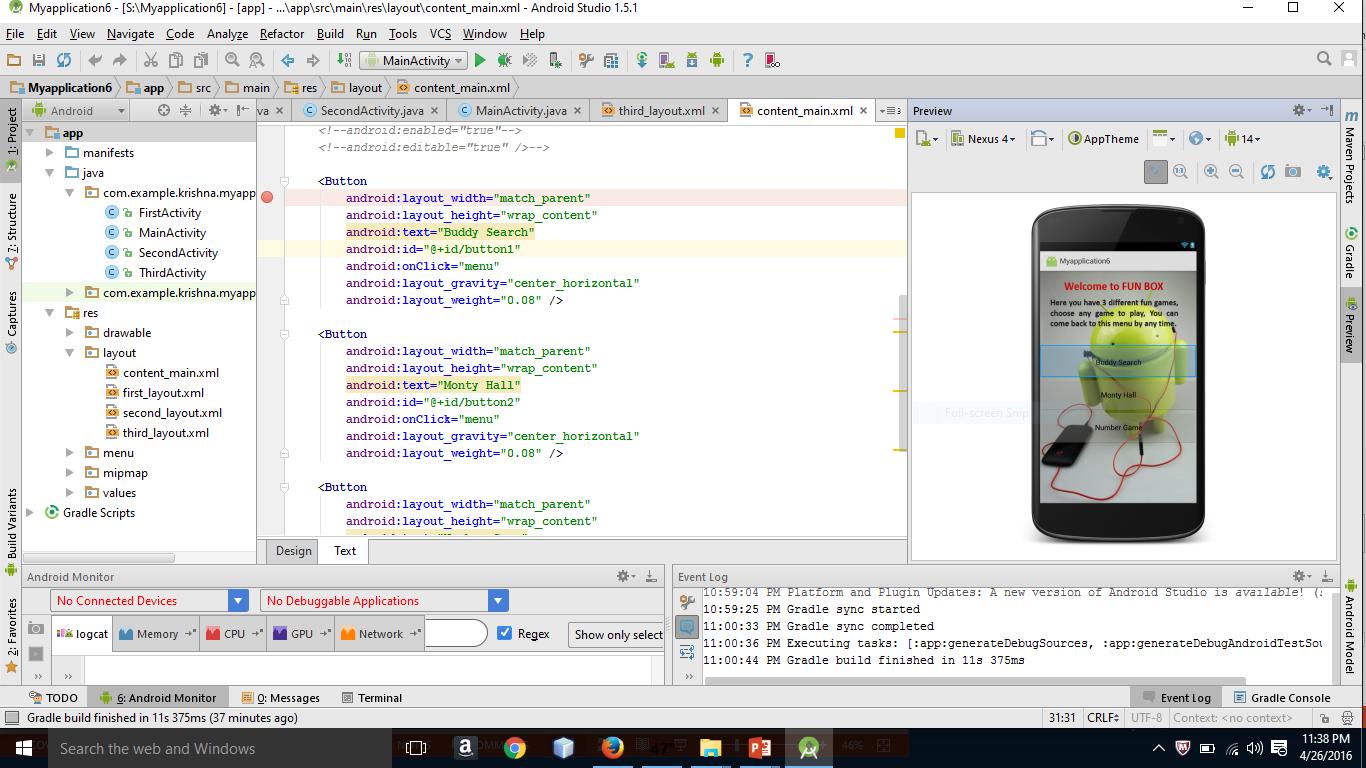
Intent

**3.4.1 Android Virtual Device (Emulator):**

AVD is the kind of mobile that we can’t touch and feel but we can run on it, which means by using this virtual emulator we can deploy any program on it, and we run it. And this helps us, and we feel everything fine when it executes successfully on emulator.

There is only thing after completion of your project is to copy the “.apk” file that generated on project / bin folder in project, and paste it on your mobile phone and installation needed.

**3.4.2 Layout manager:**



By using this layout manager we can design your own layout how it to be. For this the following xml code required.

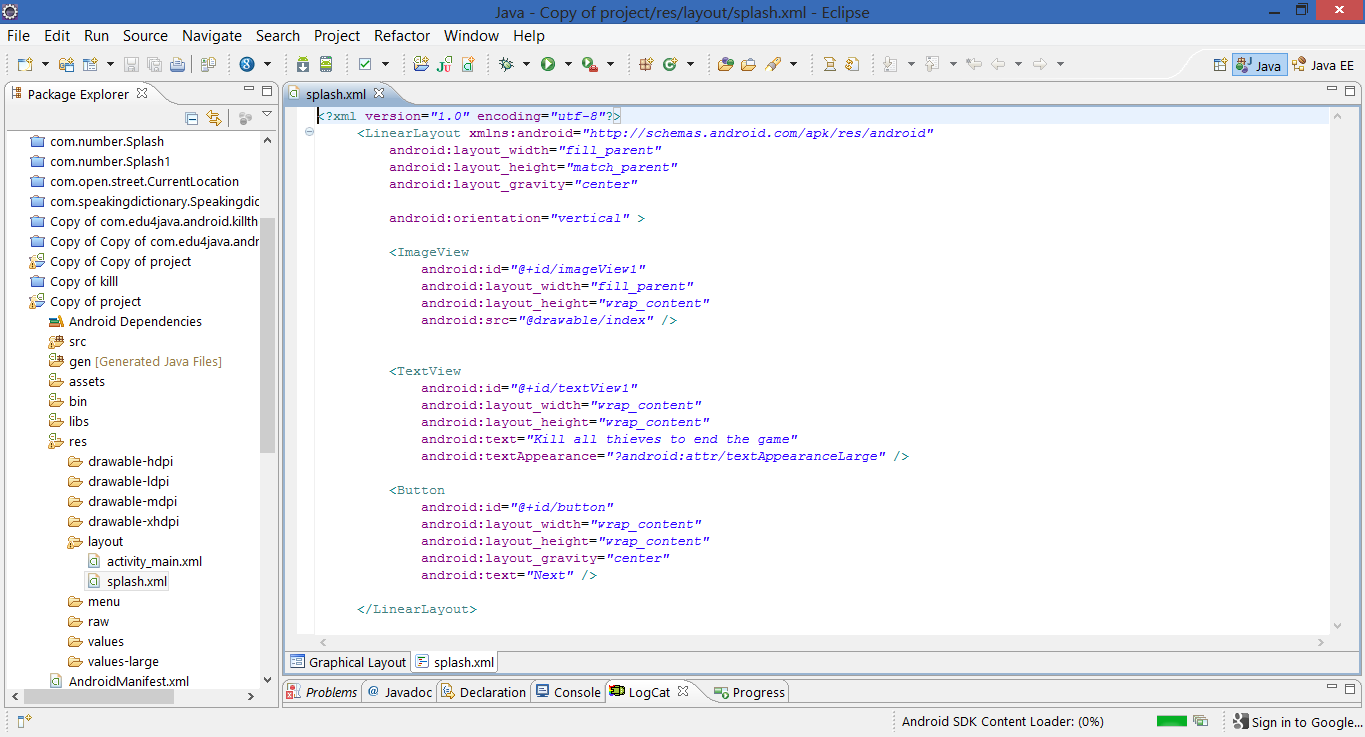
**3.4.3 Xml coding:**

XML is a king of markup language called Extensible Markup Language. By using this we can design how a user interface looks like with layouts. It takes small amount of percentage in this code.

Basically xml have some précised tags and user tags also, by using these tags a user have to determine what king of expressions he/she need. And it is the part of HTML (Hypertext Markup Language)

An example of a xml script in our gaming application is shown below

Splash.xml:



Though programming in android SDK is using java language, android has its own set of concepts that make easier to design and code for application development. This chapter briefly discusses various concepts in android that this game application has used.

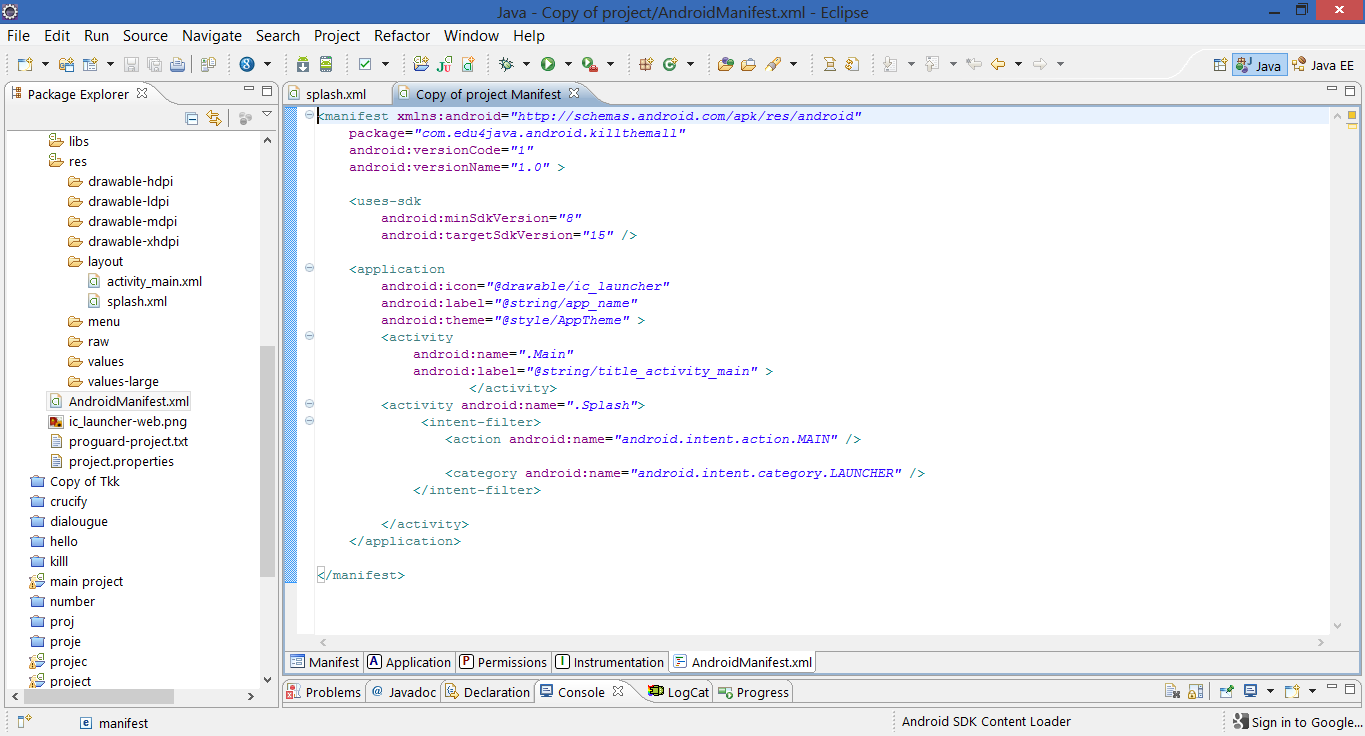
**3.4.4 Android Manifest.xml:**

An application in android must have an AndroidManifest.xml file in its development root directory. This xml file presents essential information about the application to the Android system and the information the android system should have before it can start running the application's code.

AndroidManifest.xml file contains all the other android components that our application has used within the application. The components include activities, services, broadcast receivers and Content providers. The AndroidManifest.xml file also shows which application components communicate with each other and shows which’ Intent’ and ‘Intent filter’ are tied to which application component within our application.

More importantly android manifest file specifies the list of permissions that our application would need to be installed in an android device. The permission model in android makes sure that an application that has not been granted to access few resources or services within the device indeed is prevented from accessing them.

The Androidmanifest.xml file for this game application looks like below:



## **3.4.5 Activities**

An Activity is an application component in android providing a screen with which users can interact in order to do something. The activity with some of the resource files represents the user interface in android. The android number game application contains several activities to show different screens within our application as follows:

Start Screen, Main menu, game display, scores, back to menu

## **3.4.6 Services**

A Service is an application component that can perform long-running operations in the background and does not provide a user interface. Any component within our application can start a service and it will continue to run in the background even if the user switches to another application. It does not affect the performance of the fore ground activities in the user interface as android handles the services as a separate thread by default and isolates it from the user interface. (Vogel, 2009)

This component is useful in our application to play the music in background while the user is playing game. The music is started as a service when the first screen is launched in our number game application. We will discuss later about other features of number game application to mute and unmute the music from within our application while playing game.

We don’t have any services in this project

Example services are background music or video

## **3.4.7 Intent**

Inter application communication and Intra application communication in android happen with Intent. Intent is an abstract description of an operation to be performed that can be used to launch an activity or service from the current application component. An Intent has two primary attributes namely action and data. Action represents the general action to be performed such as ACTION\_VIEW, ACTION\_EDIT etc. Data represents the information to act on when intent is raised.

There are two types of Intents namely Explicit Intents and Implicit intents. Explicit intents provide the exact component the intent has to act on and it provides the exact class to be run when the intent is raised. Implicit intents do not explicitly mention the component that can handle these intents. Any application component which has an intent filter matching the action, data category of the intent can handle the intent raised.

The Android Number Game application uses several explicit intents for different scenarios to launch activities and services.

**3.5 FRAME WORK**

Based on the requirements analysis, the application’s components, modules, interfaces and interactions of the system modules with other modules have been designed. This chapter describes the System architecture diagram, Use case diagram, Class diagram and sequence diagram of this application **[R4].**

The application is a single-player real time android game that follows a different architecture than the conventional frameworks followed in web applications. The Android OS runs on the phone and the application runs on top of the OS. The user input is a touch event captured by the application which simulates the game logic module. The audio and Animation module includes the creation of the animated sprites and production of appropriate sounds.

**USER**

**INPUT**

**ANDROID HANDSETS/ TABLETS**

**ANDROID FRAMEWORK**

**GAME**

**LOGIC**

**Audio**

**Graphics**

**O**

**U**

**T**

**P**

**U**

**T**

**G U I**

**SPRITES**

**Systems Architecture Diagram [R4].**

**3.5.1 User Input:** In our application, the user input is the touch event generated by touching the screen. The game engine monitors the onTouch event and at every touch, the corresponding coordinates are captured. If the coordinates are within the defined control areas of the device screen, the game engine would pass the control to the Game Logic module.

**3.5.2 Game Logic:** The game logic module is where the state of the game is decided based on the user inputs. The game logic includes checking any buttons and text fields to trigger the appropriate actions.

**Audio:** The audio module produces sounds based on the decisions of game logic. Different sounds are played to distinguish correct user input from an incorrect one.

**Animation:** This module is responsible for rendering the game state on the screen. Android has several facilities for graphics rendering like canvas, OpenGL, etc. Our application uses 2D canvas rendering which is refreshed every 300ms and also updated based on the user’s input.

**Score:** This module is used in saving the scores of the player. Here we didn’t use any database to save the records of individual player. It automatically updates each and every player score on the time and it doesn’t remember previous player results, we can maintain this results by using SQLite database. SQLite database is an open source database that is embedded into Android. The game logic module interacts with this module for data persistence into the tables.

**Output GUI:** This output is the resulting sound and view rendering based on the game logic.

**3.6 ANDROID STUDIO**

Android Studio is a comprehensive operating environment that packs a power of a mainframe system into user microcomputer. It provides a set of functional programs that user can use as tools to build structures and perform tasks. By using Android Studio we can import or export previously executes files, and it is possible to write any android project, by adding certain preferences on it we can create any kind of environment on it. And it is complete Android development kit, itself it has emulator and compiler and so many.

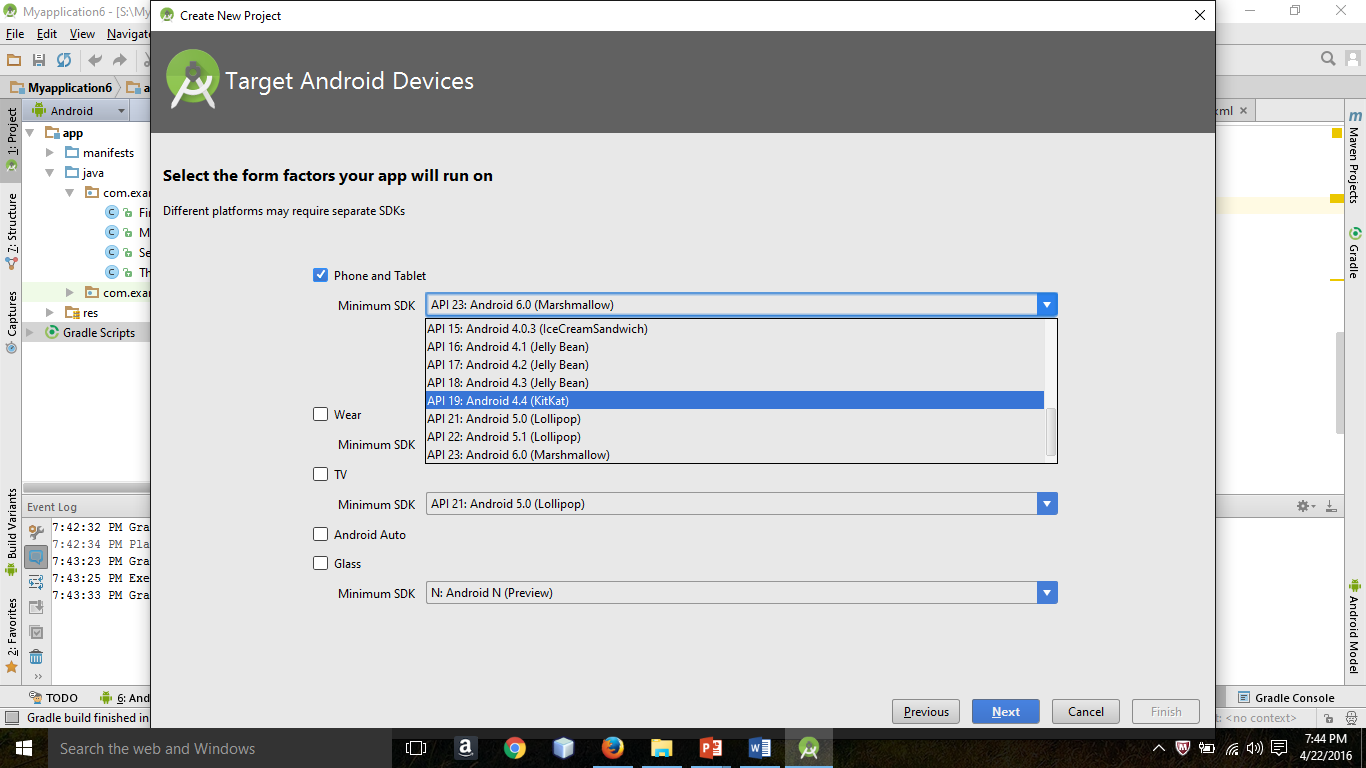
The data files contain all user data in terms of tables, index and views. The log files contain the information to open and be recovered, of undone after a transaction (Rollback). The control file physical data, media information to open and manage data files. Android Studio stores all the files and projects in particular space called workspace. We can manage this workspace from any place to any other place by browsing workspace when Studio opening.

Implementation

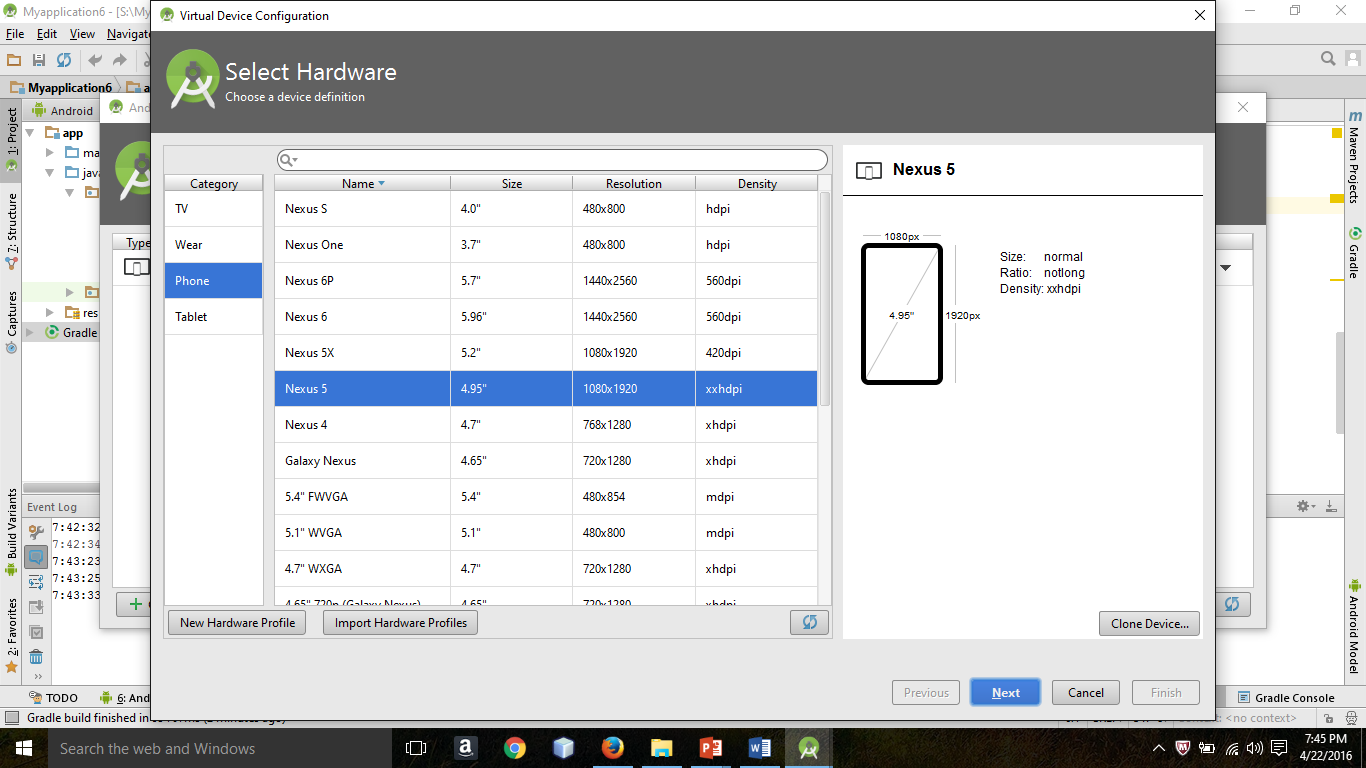
Steps Included

Installing Android Studio followed by Java Development Kit (JDK)

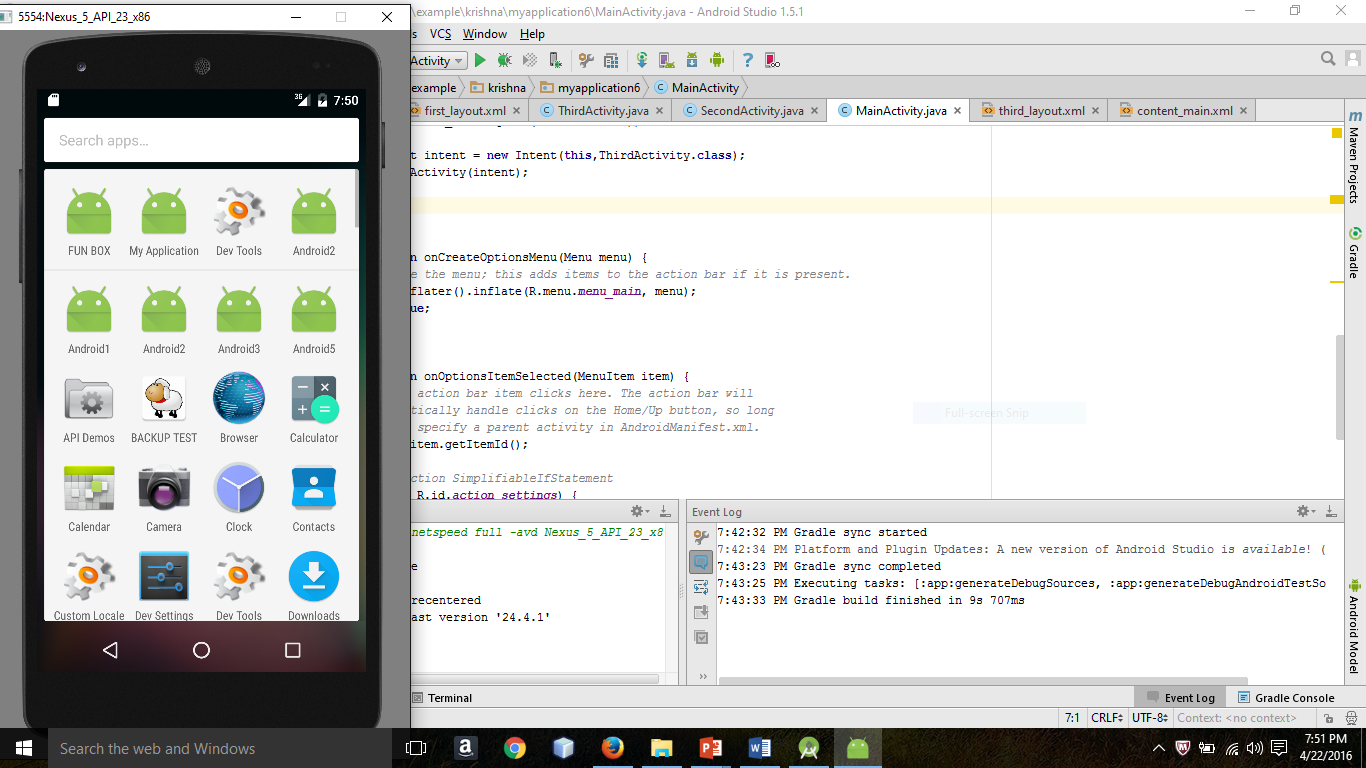
Step 1- Select version for your application

****

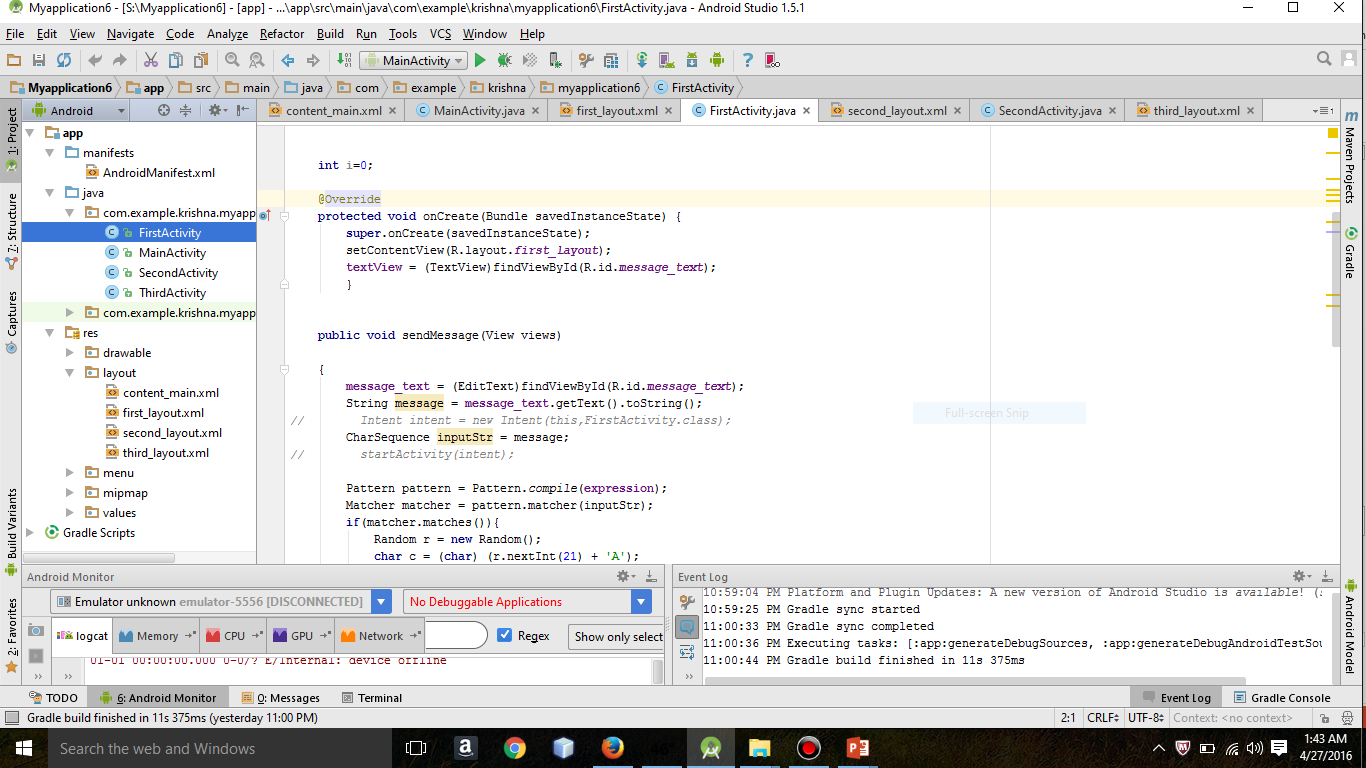
* Step 2- Select a target Android Virtual Device(AVD) to run your application



Once after you selected and run AVD appears like this:



And the below screenshot can explain how the layouts and java class files works. It’s simple whatever the layout you created in this project, just remember you have to create a java class file by referring that layout file, here first layout file is associated with fisrtActivity class file.



**4 TESTING Protocols**

**4.1 What is Fun Box:**

This document should guide the Testing Team providing the following information:

1. The main Objective of the Software

Coming to Project insight, It Is a Mobile application in Android (Name as Fun Box), the following three points can describe clearly about this project:

* It is a power packed application that itself consists small number of guessing games.
* It will have different functionality for user.
* When user enters into application Fun Box will have three different Games

Such as:

* Buddy search: it give You best friend name by taking first name as input.
* Monty hall game: it is a puzzle game that user have to choose a door that has a prize behind it among other two doors.
* Number guessing game: it gives a lucky number with quote by taking his date of birth.

**4.2 How to Compile:**

How to run it: any commands to be used to start the program

Step 1- Open project Myapplication6 in Android Studio

Step 2- This project doesn’t contain any errors and it meets all the requirements, so you won't face any problem while compiling.

Step 3- Open Android Virtual Device Manager (AVD) and select a virtual android smart phone, in which you want to run your application, and run that emulator.

Step 4- After AVD, open main java class file called MainActivity.java under the directory of app//java//com.example.krishna.myapplication6//MainActivity

Step 5- Run MainActivity(MainAtivity.java) class file to show up application

**4.3 The list, types, and format of the input data**

Input 1-Select a game from main menu consists 3 options like Buddy Search, Main Activity and Lucky Number game, Input is Button clicked type is OnclickListener

Input 2-In Buddy search and Lucky number it takes valid input as string followed by submit button

Input 3- On Monty hall it takes a door among 3 doors through buttons, input type is button clicked and method is OnclickListener

**4.4 The list, types, and format of the output data**

It gives different outputs based on application chosen by user

1-On buddy search it gives output as string in toast message

Ex: Your buddy name starts with S

2-Monty hall it gives user result whether he won or not, if he won it gives result with some rewarded amount

Ex: Congratulations you won 12$

3-Lucky Number- it gives a number in toast messages

Ex: 5 is your lucky number

**4.5 How to access any Help commands if necessary**

It shows all the instruction in every menu, so user no need to think how it works

While dealing with android studio he or she can find so many options when he clicked help on the top right of the window

On coding- after clicking alt + enter it shows list of available help options

**4.6 Sample of Data file**

In Buddy Search

Input: Robin (followed by enter button)

Output: Your buddy name starts with S (As toast message)

# **5 Testing**

Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding. Testing is the exposure of the system to trial input to see whether it produces correct output.

**5.1 Testing Phases:**

Software testing phases include the following:

Test activities are determined and test data selected.

The test is conducted and test results are compared with the expected results.

There are various types of Testing:

**5.2 Unit Testing:**

Unit testing is essentially for the verification of the code produced during the coding phase and the goal is test the internal logic of the module/program.

This project is thoroughly tested by exposing it to the various test cases regarding correct event generation, as this project passed all the tests its quality is completely assured.

**5.3 Integration Testing:**

All the tested modules are combined into sub systems, which are then tested. The goal is to see if the modules are properly integrated, and the emphasis being on the testing interfaces between the modules. On this project integration testing is done mainly while implementing menus in a sample application such as Browser for Mobiles.

**5.4 System Testing:**

It is mainly used if the software meets its requirements. The reference document for this process is the requirement document.

**5.5 Acceptance Testing:**

It is performed with realistic data of the client to demonstrate that the software is working satisfactorily.

**5.6 Testing Methods:**

Testing is a process of executing a program to find out errors. If testing is conducted successfully, it will uncover all the errors in the software. Any testing can be done basing on two ways:

**5.6 White Box Testing:**

It is a test case design method that uses the control structures of the procedural design to derive test cases. using this testing a software Engineer can derive the following test cases:

Exercise all the logical decisions on either true or false sides. Execute all loops at their boundaries and within their operational boundaries. Exercise the internal data structures to assure their validity.

**5.7 Black Box Testing:**

It is a test case design method used on the functional requirements of the software. It will help a software engineer to derive sets of input conditions that will exercise all the functional requirements of the program. Black Box testing attempts to find errors in the following categories:

Incorrect or missing functions

Interface errors

Errors in data structures

Performance errors

Initialization and termination errors

**5.8 By Black Box Testing we derive a set of test cases that satisfy the following criteria:**

Test cases that reduce by a count that is greater than one, the number of additional test cases that must be designed to achieve reasonable testing.

Test cases that tell us something about the presence or absence of classes of errors rather than errors associated only with a specific test at hand.

### Test Approach :

Testing can be done in two ways:

* Bottom up approach
* Top down approach

**5.8.1 Bottom up Approach:**

Testing can be performed starting from smallest and lowest level modules and proceeding one at a time. For each module in bottom up testing a short program executes the module and provides the needed data so that the module is asked to perform the way it will when embedded with in the larger system. When bottom level modules are tested attention turns to those on the next level that use the lower level ones they are tested individually and then linked with the previously examined lower level modules.

**5.8.2 Top down approach:**

This type of testing starts from upper level modules. Since the detailed activities usually performed in the lower level routines are not provided stubs are written. A stub is a module shell called by upper level module and that when reached properly will return a message to the calling module indicating that proper interaction occurred. No attempt is made to verify the correctness of the lower level module.

**5.9 Unit Testing**

In Unit testing each independent unit is tested separately, by isolating it from the remainder of the code to ensure parts of the code are working properly. Unit is the smallest testable part of the code, as in here the classes are treated as the base unit. Since the game application involves threading concept, it is not possible to leverage the jUnit tool for testing the individual components.

## **5.10 Compatibility Testing**

Variations in software versions, configurations, display resolutions, servers and Internet connect speeds can heavily impact the application behavior. Different specifications of devices can also make the applications to behave differently. People use different android devices and hence a good application must be 100% reliable and give best visualization effects irrespective of the device specifications. The application does not require internet hence the speed of internet is not relevant or necessary scenario in this case. To check the device compatibility, the application is tested in the both Android tablet and smart phone. The application requires Android SDK 2.3.3 or higher versions.

**5.10 Verifications & Validations**

* Start page: Application must shows initial screen
* Application: It has to produce at least 3 apps
* Access: Each application must have redirect to main menu
* Display: It has to display all 3 games clearly
* Validations: User has to enter require input
  + - System must accept numbers while taking door number in Monty hall game
    - It has to accept one string as a name while using buddy search
* Compatibility: The system must run in to any android device

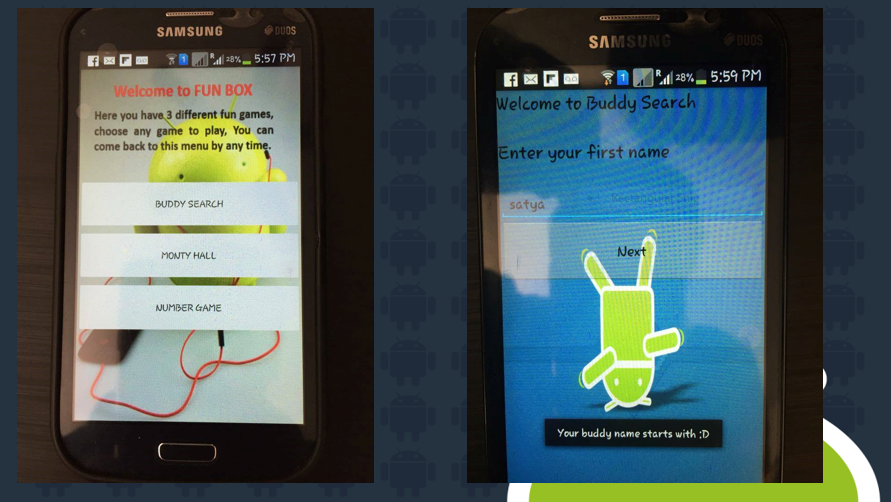
Below table clearly explains what are the project requirements and also it explains whether this project met that requirement or not, in this case this project almost met all the requirements listed below.

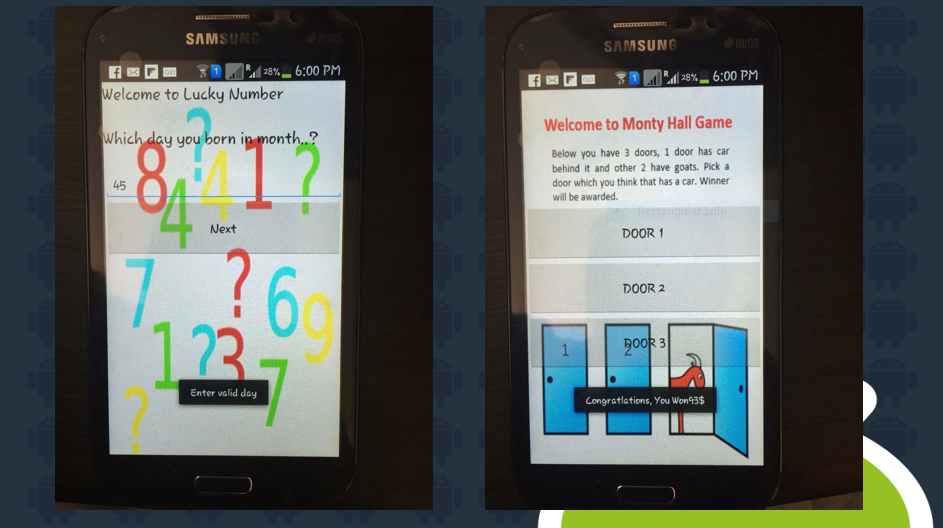
**5.11 Testing Protocol**

TRACEABILTY MATRIX

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | Team Name: Stars #6 Project Title: **FUN BOX-An Android Based Mobile Application** | | | | | | |
| Software Requirements | | Implemented | Complete | System Tested | User Acceptance Test (User Interface) | Stress Test | Performance Test |
| Requirement # 1  -Successfully Run project in Android Virtual Device Manager | | YES | YES | YES | YES | YES | YES |
| Requirement # 2  -Allow user to select favorite application with the help of buttons | | YES | YES | YES | YES | YES | YES |
| Requirement # 3  -Open selected application | | YES | YES | YES | YES | YES | YES |
| Requirement # 4  -In Buddy Search application, accepts only name(First name) in text field, otherwise show error message | | YES | YES | YES | YES | YES | YES |
| Requirement # 5  -In Monty Hall application, displays game rules & allow users to select a door among three | | YES | YES | YES | YES | YES | YES |
| Requirement # 6  -Lucky number, allows user to enter a his day of date of birth | | YES | YES | YES | YES | NO | YES |
| Requirement # 7  -Display all results once taken valid input | | YES | YES | YES | YES | YES | YES |
| Requirement # 8  -Ability to come back to main menu by anytime | | YES | YES | YES | YES | YES | YES |
| Testing Team & Names: Kareem  (SQA : Software Quality Assurance Team) | | | | | | | |

**5.12 Screen shots**:





**6 CONCLUSION**

By implementing this project one can know how to understanding the value of designing the components of overall application before implementing them.

The project also taught us application programming skills and refining the design and implementation logic of the software at every phase of the development life cycle to improve the overall performance of the application.

Mainly this app helps to create awareness on mobile application development

**7. References**

The following are the details of books and the sites which have details regarding the updated technology which is used in this project.

**Web Sites:**

1. Eli, Senior Developer.(2008). How To create sprites and random movements. <http://www.edu4java.com/en/androidgame.html>
2. Developers Guide. (2011). Retrieved September 15, 2011, from Android Developers:

<http://developer.android.com/guide/index.html>

1. Vogel, L. (2009). Android Development Tutorial. Retrieved August 2, 2011, from vogella:

<http://www.vogella.de/articles/Android/article.html>

1. A basic game architecture. (2010, July 26). Retrieved Nov 20, 2011, from Against the grain:

<http://obviam.net/index.php/2-1-a-little-about-game-architecture/>

**YouTube:**

1. YouTube, Basic animations with Sprites, a series of playlists from <https://www.youtube.com/watch?v=WxkuDwJcq6M>
2. YouTube, Android adding sounds to Sprites and user Touch events from. <https://www.youtube.com/watch?v=SV4Xom5Vljo>
3. YouTube, Android on touch listener and motion event from <https://www.youtube.com/watch?v=9d4Wua-cxZs>