**TIC-TAC-TOE(Android)**

**A SUMMER INTERN REPORT**

*Submitted by*

**TANYA GUPTA**

**Enrollment number : 04214803114**

***in partial fulfillment of Summer Internship for the award of the degree***

***of***

**BACHELOR OF TECHNOLOGY**

**IN**

**INFORMATION TECHNOLOGY**



**MAHARAJA AGRASEN INSTITUTE OF TECHNOLOGY,**

**PSP area, Sector – 22, Rohini, New Delhi – 110085**

**(Affiliated to Guru Gobind Singh Indraprastha, New Delhi)**

**Maharaja Agrasen Institute of Technology**

*To Whom It May Concern*

I, **Tanya Gupta** Enrollment No. **04214803114**, a student of Bachelors of Technology (IT), a class of 2014-18, **Maharaja Agrasen Institute of Technology, Delhi,** hereby declare that I have undergone the Summer Training in **Android Application Development** at **Coding Blocks, Delhi** under the guidance of **Mr. Arnav Gupta** from **JUNE 12th, 2016** to **AUGUST 15th**, **2016**.

I also declare that the present project report is based on the above summer training/Internship and is my original work. The content of this project report has not been submitted to any other university or institute either in part or in full for the award of any degree, diploma or fellowship.

Tanya Gupta

Enrollment No: 04214803114

INFORMATION TECHNOLOGY

5I123

**ACKNOWLEDGEMENT**

I seize this opportunity to express my indebtedness and gratitude to my supervisor for his constant support, encouragement & inspiration. This work would not have been successfully completes without his erudite guidance and remarkable patience. My profound thanks to him for his constructive criticism while allow pursuing my own idea. I am thankful for my teacher for his kind cooperation.

I express my sincere thanks and gratitude to my mentor **Mr. Arnav Gupta** and all the technical staff. I am indebted to all them for their kind help and cooperation that I received during my training.

I would like to express my deepest gratitude to my parents and all my batch mates to their kind support and constant encouragement.

Tanya Gupta

Enrollment No: 04214803114

INFORMATION TECHNOLOGY

5I123

**PREFACE**

The aim of this project to develop a prototype of a singleplayer and multiplayer **TIC-TAC-TOE** game. Tic-Tac-Toe game can be played by two players where the square block (3 x 3) can be filled with a cross (X) or a circle (O). The game will toggle between the players by giving the chance for each player to mark their move. When one of the players make a combination of 3 same markers in a horizontal, vertical or diagonal line the app will display which player has won, whether X or O.

In this app, I have implemented a 3x3 and 4x4 tic-tac-toe game. The game is designed so that a single player or two players can play tic-tac-toe using this application. The app allows you to choose grid, no. of players, symbol and even difficulty level when played with CPU. It will contain a display function and a select function to place the symbol as well as toggle between the symbols allowing each player a turn to play the game. It will update after each player makes their move and check for the conditions of game as it goes on. Overall app works without any bugs and is able to use.

It is a simple android application that requires no pre-requisite knowledge for a user.

**CONTENTS**

|  |  |
| --- | --- |
| **1. ANDROID** | **9** |

* 1. Introduction
  2. Why is Android better?
  3. Android Version History
  4. Features & Specifications
  5. Android Architecture
  6. Building blocks for applications
     1. Activity
     2. Services
     3. Intent
     4. Broadcast Receivers
     5. Content Providers
     6. Fragment
     7. AndroidManifest.xml
  7. Categories of Android Applications

|  |  |
| --- | --- |
| 1. **Integrated Development Environment (IDE)** | **18** |

2.1 Features:

2.2 Project Structure

2.3 The User Interface

|  |  |
| --- | --- |
| 1. **Properties** | **21** |

3.1 Security

3.2 Merits

1. Android Market

2. Customization

3. Rooting

4. The Android Community

3.3. Demerits

3.4 Limitations

|  |  |
| --- | --- |
| 1. **Project Profile** | **24** |

4.1 Problem Definition

4.2 System requirements

a. Project Profile

b. Project Tools

c. Software Requirements

d. Hardware Requirements

|  |  |
| --- | --- |
| 1. **Analysis** | **26** |

5.1 Existing system

5.2 Feasibility Study

• Operational Feasibility

• Technical Feasibility

• Economical Feasibility

5.3 Proposed system definition

5.4 Features of proposed system

|  |  |
| --- | --- |
| 1. **Software Interface** | **28** |

* 1. Main Menu
  2. Play Now Option
* GRID SIZE
* DIFFICULTY LEVEL
* NO. OF PLAYERS
* Playing against CPU
* Playing against human
  1. Buy Pro Pack
  2. High Score

|  |  |
| --- | --- |
| 1. **Future Enhancement** | **35** |
| 1. **Conclusion** | **36** |
| 1. **Bibliography** | **37** |

**LIST OF FIGURES**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No.** | **Figure No.** | **Screenshot Description** | **Page Number** |
| 01 | Figure 1 | Android | 9 |
| 02 | Figure 2 | Commonly used mobile OS | 10 |
| 03 | Figure 3 | Android Evolution | 12 |
| 04 | Figure 4 | Features of android OS | 12 |
| 05 | Figure 5 | Android Architecture | 13 |
| 06 | Figure 6 | Building block | 14 |
| 07 | Figure 7 | Activity Cycle | 15 |
| 08 | Figure 8 | Categories | 17 |
| 09 | Figure 9 | The project files in Android view | 19 |
| 10 | Figure 10 | The Android Studio main window | 20 |
| 11 | Figure 11 | Android Market | 22 |
| 12 | Figure 12 | Customization | 22 |
| 13 | Figure 13 | Rooting | 22 |
| 14 | Figure 14 | Android Community | 23 |

**LIST OF SCREENSHOT**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No.** | **Screenshot No.** | **Screenshot Description** | **Page Number** |
| 01 | Screenshot 1 | Main Menu | 28 |
| 02 | Screenshot 2 | Play Now Settings | 29 |
| 03 | Screenshot 3 | Changed Settings | 29 |
| 04 | Screenshot 4 | Grid Size(3 X 3) | 30 |
| 05 | Screenshot 5 | Grid Size(4 X 4) | 30 |
| 06 | Screenshot 6 | Dialog box when player wins against cpu | 30 |
| 07 | Screenshot 7 | Dialog box when player lose against cpu | 30 |
| 08 | Screenshot 8 | Playing against CPU | 31 |
| 09 | Screenshot 9 | Playing against Human | 32 |
| 10 | Screenshot 10 | Buy Pro Pack | 33 |
| 11 | Screenshot 11 | High Scores | 34 |

**LIST OF TABLES**

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Table Description** | **Page Number** |
| 01 | Android Version History | 11 |

**1. ANDROID**

World is contracting with the growth of mobile phone technology. As the number of users is increasing day by day, facilities are also increasing. Starting with simple regular handsets which were used just for making phone calls, mobiles have changed our lives and have become part of it. Now they are not used just for making calls but they have innumerable uses and can be used as a Camera , Music player, Tablet PC, T.V. , Web browser etc . And with the new technologies, new software and operating systems are required.

**1.1 Introduction**

[Operating Systems](http://www.engineersgarage.com/articles/operating-systems-tutorial) have developed a lot in last 15 years. Starting from black and white phones to recent smart phones or mini computers, mobile OS has come far away. Especially for smart phones, Mobile OS has greatly evolved from Palm OS in 1996 to Windows pocket PC in 2000 then to Blackberry OS and Android.

One of the most widely used mobile OS these days is ANDROID. **Android** is a software bunch comprising not only operating system but also middleware and key applications. Android Inc was founded in Palo Alto of California, U.S. by Andy Rubin, Rich miner, Nick sears and Chris White in 2003. Later Android Inc. was acquired by Google in 2005. After original release there have been number of updates in the original version of Android.

Android has the largest installed base of all operating systems of any kind. Android has been the best-selling OS on tablets since 2013, and on smartphones it is dominant by any metric. Initially developed by Android, Inc., which Google bought in 2005, Android was unveiled in 2007 along with the founding of the Open Handset Alliance – a consortium of hardware, software, and telecommunication companies devoted to advancing open standards for mobile devices. As of July 2013, the Google Play store has had over one million Android applications published – including many "business-class apps” that rival competing mobile platforms – and over 50 billion applications downloaded. An April – May 2013 survey of mobile application developers found that 71% of developers create applications for Android, and a 2015 survey found that 40% of full-time professional developers see Android as their priority target platform, which is comparable to Apple's iOS on 37% with both platforms far above others. In September 2015, Android had 1.4 billion monthly active users.

Fig. 1: Android

Android is popular with technology companies that require a ready-made, low-cost and customizable operating system for high-tech devices. The success of Android has made it a target for patent (and copyright) litigation as part of the so-called "smartphone wars" between technology companies.

**1.2** **Why is Android better?**

There are many advantages to developing for the Android platform:

* **Google Applications:** Android includes most of the time many Google applicationslike Gmail, YouTube, and Maps etc. These applications are delivered with the machine most of the time, except in certain cases, such as some phones running Android on which the provider has replaced Google applications by its own applications.

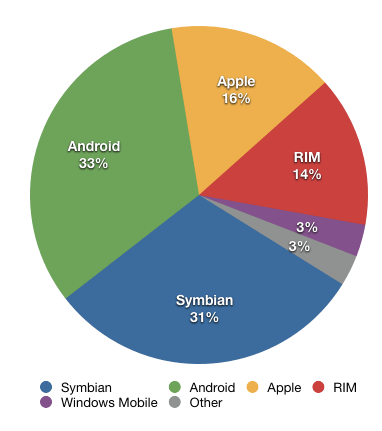
* **Widgets:** With android, it is possible to use small tools that can most often getinformation. These widgets are directly visible on the main window.
* **Android Market:** This is an online software store to buy applications. Developerswho created applications can add them into the store, and these applications can be downloaded by users, they can be both free and paid.
* **Multitasking:** Android allows multitasking in the sense that multiple applications canrun simultaneously. With Task Manager it is possible view all running tasks and to switch from one to another easily.
* **SDK:** A development kit has been put at disposal of everybody. Accordingly, anydeveloper can create their own applications, or change the android platform. This kit consists a set of libraries, powerful tools for debugging and development, a phone emulator, thorough documentation, FAQs and tutorials.
* **Modifiability:** This allows everyone to use, improve or transform the function ofAndroid, for example, transform the interface in functions of uses, to transform the platform in a real system embedded Linux.

Fig. 2: Commonly used mobile OS

**1.3 Android Version History**

The history and versions of android are interesting to know. The code names of android ranges from A to J currently, such as Aestro, Blender, Cupcake, Donut, Eclair, Froyo, Gingerbread, Honeycomb, Ice Cream Sandwitch,Jelly Bean and KitKat. Let's understand the android history pointy:

* Initially, **Andy Rubin** founded Android Incorporation in Palo Alto, California, United States in October, 2003.
* In 17th August 2005, Google acquired android Incorporation. Since then, it is in the subsidiary of Google Incorporation.
* The key employees of Android Incorporation are **Andy Rubin**, **Rich Miner**, **Chris White** and **Nick Sears**.

Table 1: Android Version History

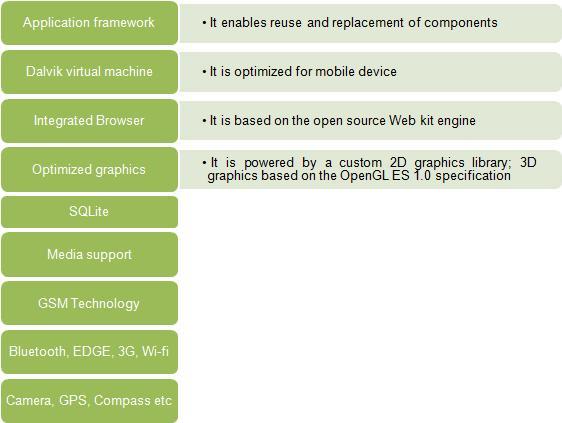
|  |  |  |  |
| --- | --- | --- | --- |
| **Code name** | **Version Number** | **Release Date** | **API Level** |
| N/A | 1.0 | 23rd Sept, 2008 | 1 |
| N/A | 1.1 | 9th Feb, 2009 | 2 |
| Cupcake | 1.5 | 27th April, 2009 | 3 |
| Donut | 1.6 | 15th Sept, 2009 | 4 |
| Éclair | 2.0 - 2.1 | 26th Oct, 2009 | 5 - 7 |
| Froyo | 2.2 – 2.2.3 | 20th May, 2010 | 8 |
| Gingerbread | 2.3 – 2.3.7 | 6th Dec, 2010 | 9 - 10 |
| HoneyComb | 3.0 - 3.2.6 | 22nd Feb, 2011 | 11 - 13 |
| Ice Cream Sandwich | 4.0 – 4.0.4 | 18th Oct, 2011 | 14 - 15 |
| Jelly Bean | 4.1 – 4.3.1 | 9th July, 2012 | 16 - 18 |
| KitKat | 4.4 – 4.4.4 | 31st Oct, 2013 | 19 – 20 |
| Lollipop | 5.0 – 5.1.1 | 12th Nov, 2014 | 21 - 22 |
| Marshmallow | 6.0 – 6.0.1 | 5th Oct, 2015 | 23 |
| Nougat | 7.0 | 22nd Aug, 2016 | 24 |

Fig. 3: Android Evolution 

**1.4 Features & Specifications**

**Android** is a powerful Operating System supporting a large number of applications in [Smart Phones](http://www.engineersgarage.com/articles/smart-phones). These applications make life more comfortable and advanced for the users. Hardware that support Android are mainly based on [ARM architecture](http://www.engineersgarage.com/articles/arm-advanced-risc-machines-processors) platform. Some of the current features and specifications of android are:

Fig. 4: Features of android OS



**1.5 Android Architecture**

Android Architecture or Android Software Stack is categorized into five parts:

1. **Linux Kernel:** It is the heart of Android architecture that exists at the root of Androidarchitecture. Linux kernel is responsible for device drivers, power management, memory management, device management and resource access.
2. **Native Libraries:** On the top of Linux kernel, there are Native libraries such asWebKit, OpenGL, FreeType, SQLite, Media, C runtime library etc.

The Webkit library is responsible for browser support, SQLite is for database, FreeType font support, Media for playing and recording audio and video formats.

1. **Android Runtime:** In Android runtime, there are core libraries and DVM (DalvikVirtual Machine) which is responsible to run android application. DVM is like JVM but it is optimized for mobile devices. It consumes less memory and provides fast performance.
2. **Android Framework:** On the top of Native libraries and android runtime, there isandroid framework. Android framework includes Android API’s such as UI (User Interface), telephony, resources, locations, Content Providers (data).
3. **Applications:** On the top of android framework, there are applications.
4. **Hardware abstraction layer (HAL):** The hardware abstraction layer (HAL) defines a standard interface for hardware vendors to implement and allows Android to be agnostic about lower-level driver implementations. The HAL allows you to implement functionality without affecting or modifying the higher level system. HAL implementations are packaged into modules (.so) file and loaded by the Android system at the appropriate time.



Fig. 5: Android Architecture

# 1.6 Building blocks for applications

Application components are the essential building blocks of an Android application. These components are loosely coupled by the application manifest file AndroidManifest.xml that describes each component of the application and how they interact.

An android component is simply a piece of code that has a well defined life cycle e.g. Activity, Receiver, Service etc.

There are following four main components that can be used within an Android application:

• An activity is a class that represents a single screen. It is
like a Frame in AWT.Activity
• A view is the UI element su...

Fig. 6: Building block

**1.6.1 Activity**

An activity represents a single screen with a user interface, in-short Activity performs actions on the screen. For example, an email application might have one activity that shows a list of new emails, another activity to compose an email, and another activity for reading emails. If an application has more than one activity, then one of them should be marked as the activity that is presented when the application is launched.



Fig. 7: Activity Cycle

#### 1.6.2 Services

#### A service is a component that runs in the background to perform long-running operations. For example, a service might play music in the background while the user is in a different application, or it might fetch data over the network without blocking user interaction with an activity.

**1.6.3 Intent**

**Android Intent** is the *message* that is passed between components such as activities, content providers, broadcast receivers, services etc.

It is generally used with startActivity() method to invoke activity, broadcast receivers etc.

The **dictionary meaning** of intent is *intention or purpose*. So, it can be described as the intention to do action.

The Labelled Intent is the subclass of android.content.Intent class.

Android intents are mainly used to:

* Start the service
* Launch an activity
* Display a web page
* Display a list of contacts
* Broadcast a message
* Dial a phone call etc.

## Types of Android Intents

There are two types of intents in android: implicit and explicit :

### *1) Implicit Intent*

**Implicit Intent** doesn't specify the component. In such case, intent provides information of available components provided by the system that is to be invoked.

### *2) Explicit Intent*

**Explicit Intent** specifies the component. In such case, intent provides the external class to be invoked.

**1.6.4 Broadcast Receivers**

Broadcast Receivers simply respond to broadcast messages from other applications or from the system. For example, applications can also initiate broadcasts to let other applications know that some data has been downloaded to the device and is available for them to use, so this is broadcast receiver who will intercept this communication and will initiate appropriate action.

A broadcast receiver is implemented as a subclass of BroadcastReceiver class and each message is broadcaster as an Intent object.

**1.6.5 Content Providers**

A content provider component supplies data from one application to others on request. Such requests are handled by the methods of the ContentResolver class. The data may be stored in the file system, the database or somewhere else entirely.

A content provider is implemented as a subclass of ContentProvider class and must implement a standard set of APIs that enable other applications to perform transactions.

#### 1.6.6 Fragment

Fragments are like parts of activity. An activity can display one or more fragments on the screen at the same time.

#### 1.6.7 AndroidManifest.xml

It contains informations about activities, content providers, permissions etc. It is like the web.xml file in Java EE.

**1.7 Categories of Android Applications**

There are many android applications in the market. The top categories are:

* Entertainment
* Communication
* Productivity
* Personalization
* Music and Audio
* Social
* Media and Video
* Travel and Local etc.

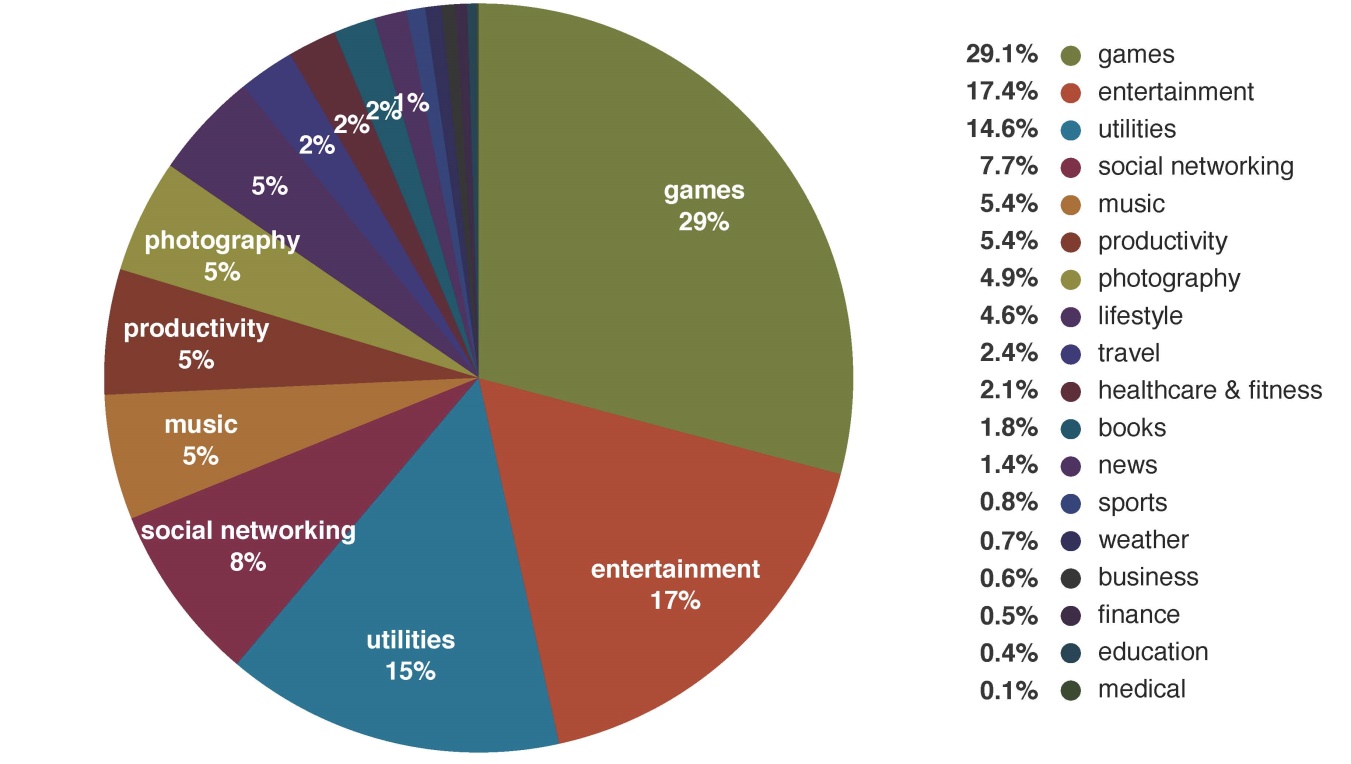


Fig. 8: Categories

2.**Integrated Development Environment (IDE)**

**Android software development** is the process by which new applications are created forthe Android operating system. Applications are usually developed in Java programming language using the Android software development kit (SDK), but other development environments are also available.

****

**Android Studio**

Android Studio is the official IDE for Android development, and with a single download includes everything you need to begin developing Android apps:

* IntelliJ IDE + Android Studio plugin
* Android SDK Tools
* Android Platform-tools
* A version of the Android platform
* Android Emulator with an Android system image including Google Play Services

It was announced on May 16, 2013 at the Google I/O conference. Android Studio is freely available under the Apache License 2.0.

Android Studio was in early access preview stage starting from version 0.1 in May 2013, then entered Beta stage starting from version 0.8, which was released in June 2014. The first stable build was released in December 2014, starting from version 1.0.

Based on JetBrain’s IntelliJ IDEA software, Android Studio is designed specifically for Android development. It is available for download on Windows, Mac OSX and Linux, and replaced Eclipse Android Development Tools (ADT) as Google’s primary IDE for native Android development.

**2.1 Features:**

* A flexible Gradle-based build system
* A fast and feature-rich emulator
* A unified environment where you can develop for all Android devices
* Instant Run to push changes to your running app without building a new APK
* Code templates and GitHub integration to help you build common app features and import sample code
* Extensive testing tools and frameworks
* Lint tools to catch performance, usability, version compatibility, and other problems
* C++ and NDK support.

## 2.2 Project Structure

Each project in Android Studio contains one or more modules with source code files and resource files. Types of modules include:

* Android app modules
* Library modules
* Google App Engine modules

All the build files are visible at the top level under **Gradle Scripts** and each app module contains the following folders:

* **manifests**: Contains the AndroidManifest.xml file.
* **java**: Contains the Java source code files, including JUnit test code.
* **res**: Contains all non-code resources, such as XML layouts, UI strings, and bitmap images.

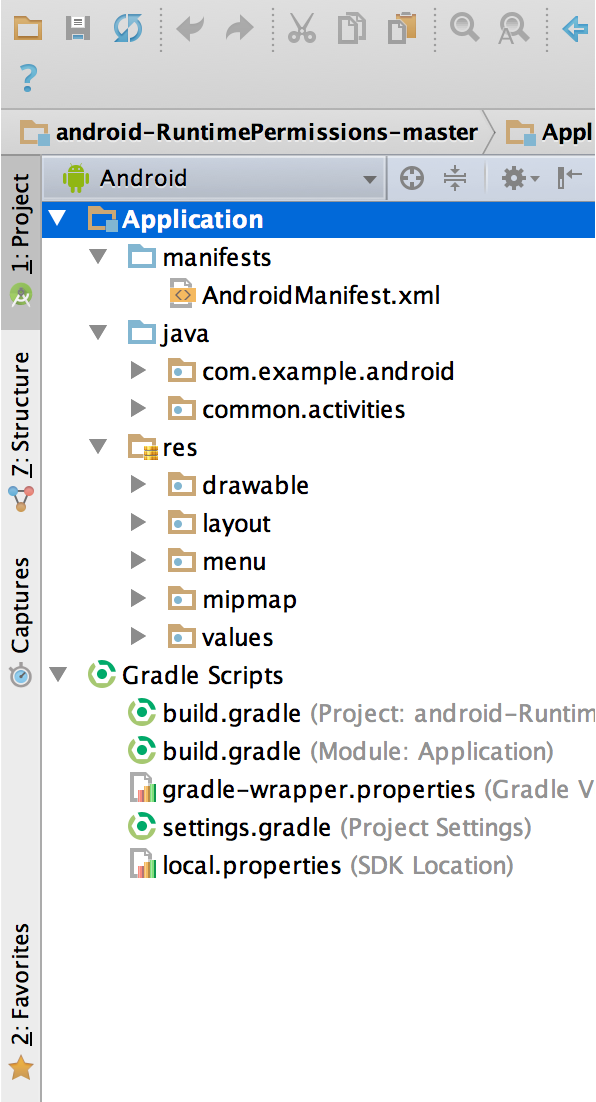


Fig. 9: The project files in Android view

**2.3 The User Interface**

The Android Studio main window is made up of several areas identified in the given figure:

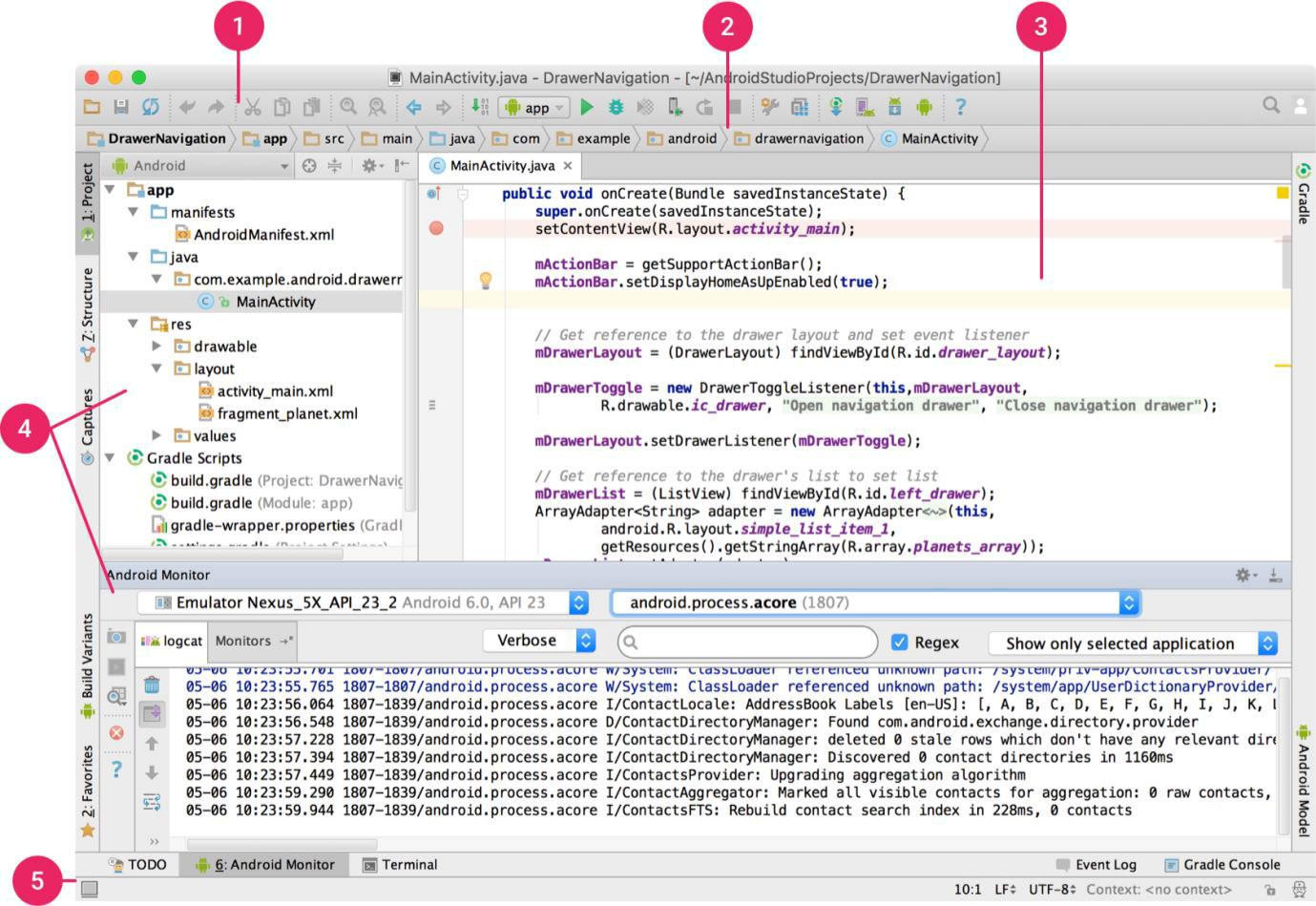


Fig. 10: The Android Studio main window

1. The **toolbar** lets you carry out a wide range of actions, including running your app and launching Android tools.
2. The **navigation bar** helps you navigate through your project and open files for editing. It provides a more compact view of the structure visible in the Project tool window.
3. The **editor window** is where you create and modify code. Depending on the current file type, this window can change. For example, when viewing a layout file, the editor window displays the layout editor and offers the option to view the corresponding XML file.
4. **Tool windows** give you access to specific tasks like project management, search,version control, and more. You can expand them and collapse them.
5. The **status bar** displays the status of your project and the IDE itself, as well as any warnings or message.

**3. Properties**

**3.1 Security:**

* Android is a multi-process system, in which each application (and parts of the system) runs in its own process. Most security between applications and the system is enforced at the process level through standard Linux facilities, such as user and group IDs that are assigned to applications.
* Android is designed having multilayer security which provides flexibility for this platform. When attackers attempt attack on device, android platform help to reduce the portability of the attack. There are key components of android security which are described as follows:
* Design review:-when a security model is designed then it will be reviewed by the developers so that risk level will be very less while using the model.
* Code review and penetrating testing:-the goal of this code review is that in which it will be checked that how the system will become strong?
* Open source and community review:-android uses open source technologies that have significant external review such as Linux kernel.
* Incident response:-android team enables the rapid mitigation of vulnerabilities to ensure that potential risks to all android users are minimized.

**3.2** **Merits:**

* The ability for anyone to customize the Google Android platform
* It gives you better notification.
* It lets you choose your hardware.
* It has better app market(1,80,000 application)
* A more mature platform
* With the support of many applications, the user can change the screen display.
* With Google chrome you can open many window at once.
* Supports all Google services: Android operating system supports all of Google services ranging from Gmail to Google reader. all Google services can you have with one operating system, namely Android.

**1. Android Market:**

[](http://www.geekandblogger.com/wp-content/uploads/2012/07/Android-Market.jpg)Despite it being a relatively new application market, android market is the fastest growing market in the world. You can get almost any software you wish for in the android market. The ready availability of software to run on your android phone means that your phone will have more functionality and thus making it convenient to use. What makes this even better is the fact that anyone can place any application or program they create into the market. This includes even you. This is because of the fact that android is basically an open source software. It makes customizing your phone easier, and thus guaranteeing you the best experience that any phone can give.

Fig. 11: Android Market

[](http://www.geekandblogger.com/wp-content/uploads/2012/07/Android-Customization.jpg)**2. Customization:**

Android offers the best in terms of end-user experience mainly because of the extent that one can customize his phone or device. The secret to its ability to be customized lies in the fact that it is an open source software. This means that anybody with programming skills can create applications for his or her device. This creates a limitless field for customization in terms of color, design, functionality and general appearance of your phone or tablet. This makes it almost impossible for you to miss what really works for you.

Fig. 12: Customization

**3. Rooting:**

[](http://www.geekandblogger.com/wp-content/uploads/2012/07/Android-Rooting.jpg)Rooting further expands the scope of functionality of your phone and boosts its performance multiple times over. Rooting makes it possible for you to change system files in your device and thus enabling you to further customize it for a better experience. Through this process, you will be able to install a customized and altered android operating system and thus enabling you to unleash the full benefits of your device. You can then not only increase its speed and functionality, but also change the inbuilt color schemes and general appearance of your phone. In case you are a developer or you have one helping you out, it allows you to do almost anything with your android phone. The only downside to rooting is the fact that it is against the terms of warranty.  However, considering the things that it can make possible with your phone, it is worth the risk.

Fig. 13: Rooting

[](http://www.geekandblogger.com/wp-content/uploads/2012/07/Android-Community.jpg)**4. The Android Community:**

This is one of the major reasons as to why choosing an android device is the best decision you will ever make. This is because the large android community provides you with the necessary help and support to help you get the best out of your device. The countless forums, blogs, websites and guides written by experts will go a long way in boosting your experience as a user.

Fig. 14: Android Community

**3.3.** **Demerits:**

* Android Market is less control of the manager, sometimes there are malware.
* Wasteful Batteries, This is because the OS is a lot of "process" in the background causing the battery quickly drains.
* Sometimes slow device company issued an official version of Android your own .
* Extremely inconsistence in design among apps. Very unstable and often hang or crash.

**3.4 Limitations:**

* Development requirements in
* Java
* Android SDK
* Eclipse IDE (optional)
* Bluetooth limitations:-
* Android doesn't support:
* Bluetooth stereo
* Contacts exchange
* Modem pairing
* Wireless keyboards
* Firefox mobile isn't coming to android because of android limitations Apps in Android Market need to be programmed with a custom form of Java → Mozilla and the Fennec won't have that
* There are no split or interval times available.
* Small memory size.
* Continuous Internet connection is required.

1. **Project Profile:**

**4.1 Problem Definition:**

Games provide a real source of enjoyment in daily life. Games also are helpful in improving the physical and mental health of human. Apart from daily life physical games, people also play computer games. These games are different than those of physical games in a sense that they do not involve much physical activity rather mental and emotional activities. Getting games to react back to the user of a game has always been long hard question for game programmers. Because, lets just face it, a good game that doesn’t challenge the user’s ability to play the game doesn’t keep the user around very long. This idea can be applied to any form of game that is out there. Board games are never fun when the opponent that he or she is playing doesn’t learn or catches on. With today’s computers always advancing, programmers are always looking for new ways to make a video game more interesting and challenging for the user.

Tic-Tac-Toe game [1] can be played by two players where the square block (3 x 3) can be filled with a cross (X) or a circle (O). The game will toggle between the players by giving the chance for each player to mark their move. When one of the players make a combination of 3 same markers in a horizontal, vertical or diagonal line the program will display which player has won, whether X or O. The Tic-Tac-Toe game is most familiar among all the age groups. The friendliness of Tic-tac-toe games makes them ideal as a pedagogical tool for teaching the concepts of good sportsmanship. The game is a very good brain exercise. It involves looking ahead and trying to figure out what the person playing against you might do next.

**4.2 System requirements:**

1. **Project Profile:**

|  |  |
| --- | --- |
| Project Title: | **Tic-Tac-Toe** |
| Developed by: | Tanya Gupta |
| Duration: | June, 2016 – July, 2016. |
| Internal Guide: | Mr. Arnav Gupta |

1. **Project Tools:**

|  |  |
| --- | --- |
| Front-end Tools: | Android Studio |
| Back-end Tools: | MySQL Server |
| Platform: | Android SDK |

1. **Software Requirements:**

|  |  |
| --- | --- |
| Application Front-end Tools: | Android Studio |
| Database Back-end Tools: | MySQL Server |
| Office Automation Tools: | Microsoft office 2010 |

1. **Hardware Requirements:**

|  |  |
| --- | --- |
| Application Installation | Android enabled GSM/CDMA phone |
| Internet Connection | Required |
| Version | 1.0 |
| Memory | 2.12 MB |

1. **Analysis:**

**5.1 Existing system :**

Tic-Tac-Toe (or Noughts and crosses, Xs and Os) is a pencil-and-paper game for two players, O and X, who take turns marking the spaces in a 3 x 3 grid. The player who succeeds in placing three respective marks in a horizontal, vertical or diagonal row wins the game.

**5.2 Feasibility Study:**

* + **Operational Feasibility:**

By feasibility study we mean study of the current operational systems and brief consideration of alternative methods computerizing these tasks. The purpose of the feasibility study is to investigate the present system, evaluate the possible application of computer-based methods, select a tentative system, evaluate the cost and effectiveness of the proposed system, evaluate the impact of the proposed system on existing personal and ascertain the need for new personnel.

From the user point of view, a Tic-Tac-Toe here is easy to operate because it only uses simple step by step to perform any task. The application is simple for smart phone users. So there is no need to give any instruction about how to use these systems. There is no need to learn any special command and training to use this system.

* + **Technical Feasibility:**

Technical feasibility refers to the ability of the process to take advantages of the current state of the technology in pursuing further improvement. The front end tool proposed is easily compatible with the current hardware configuration in the organization. The back end tool proposed has the capacity to hold the data required for using the new system.

Tic-Tac-Toe uses the existing technology only. It uses internet for fetching the data of high scores. Now a day these technologies are improved day by day. So it’s convenient to us use these technology. So there is no need to develop any hardware to provide these facilities. Tic-Tac-Toe is run on android operating system which is available in most of smart phone devices. So there is no need to install any software or driver to run this application.

* + **Economical Feasibility:**

For declaring that the system is economically feasible, system will be cost effective and budgetary constraints, it should be cheap and quick. There isn’t any extra requirement of peripheral or software for development of system as it can be completed with the available resource.

**5.3 Proposed system definition:**

In this app, I have implemented a 3x3 and 4x4 tic-tac-toe game. The game is designed so that a single player or two players can play tic-tac-toe using this application. The app allows you to choose grid, no. of players, symbol and even difficulty level when played with CPU. It will contain a display function and a select function to place the symbol as well as toggle between the symbols allowing each player a turn to play the game. It will update after each player makes their move and check for the conditions of game as it goes on. Overall app works without any bugs and is able to use.

**5.4 Features of proposed system:**

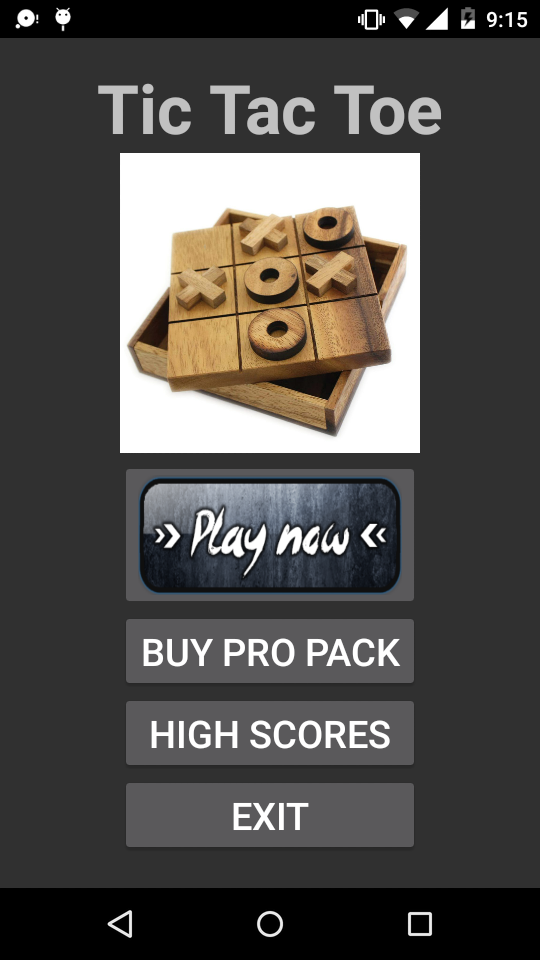
The computer implementation of the game Tic-Tac-Toe has many features as compared to the traditional way of playing it with paper and pencil. The various features are:

* The game has been made user friendly with proper use of android studio.
* The user can play as many games without any interruption.
* The user can choose any symbol he/she wants to.
* The game has been made as a thorough expert system.
* The player can either win the game, draw the game or will lose the game.
* It’s a good brain exercise for all age group people.

1. **Software Interface:**
   1. **Main Menu**

Main menu provides user ability to choose any of the following options:-

* Play Now
* Buy Pro Pack
* High Scores
* Exit

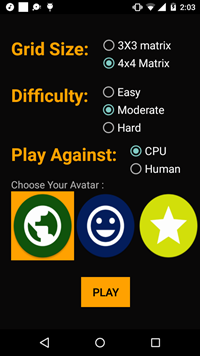
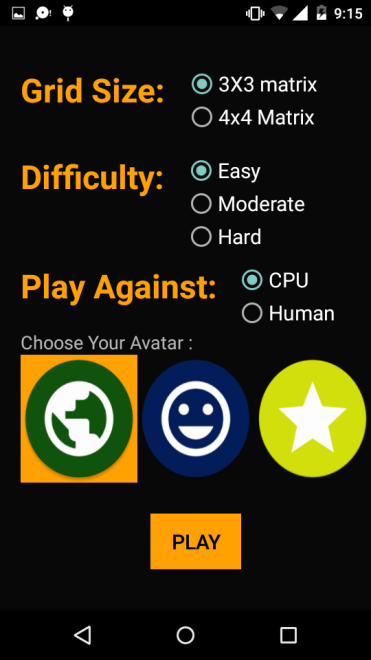


Screenshot 1: Main Menu

* 1. **Play Now Option**

When the user clicks on play now then it provides user ability to change:-

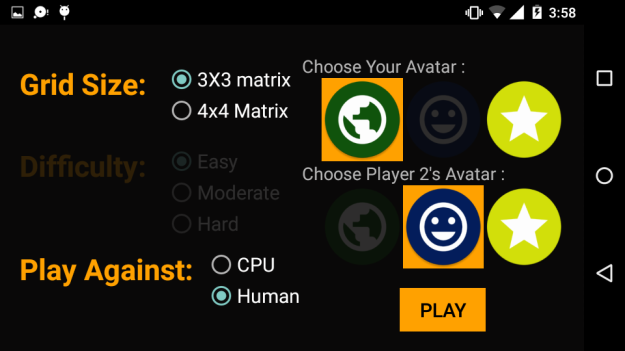
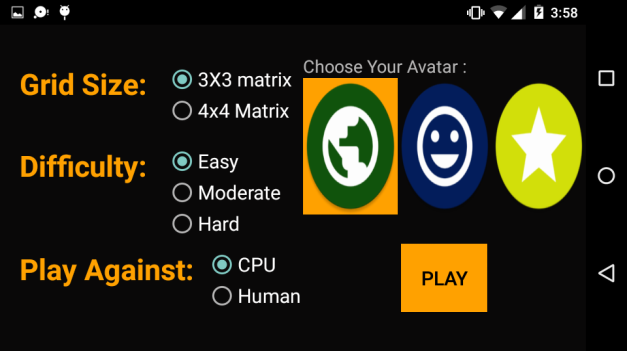
* Grid Size
* Difficulty level
* Number of players



Screenshot 2: Play Now Settings Screenshot 3: Changed Settings

This is the default Settings that appear on screen. Now, user can click on radio Buttons and choose any option.

*Landscape View:*

**

* + **GRID SIZE**

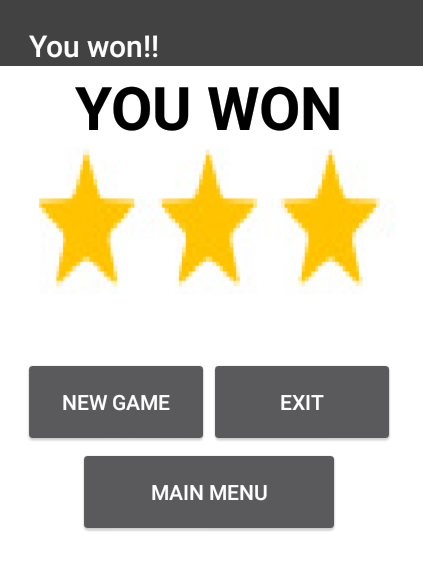
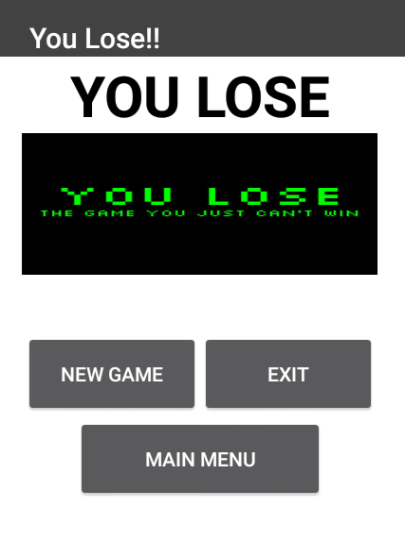
Screenshot 4: Grid Size(3 X 3) Screenshot 5: Grid Size(4 X 4)

* + **DIFFICULTY LEVEL**

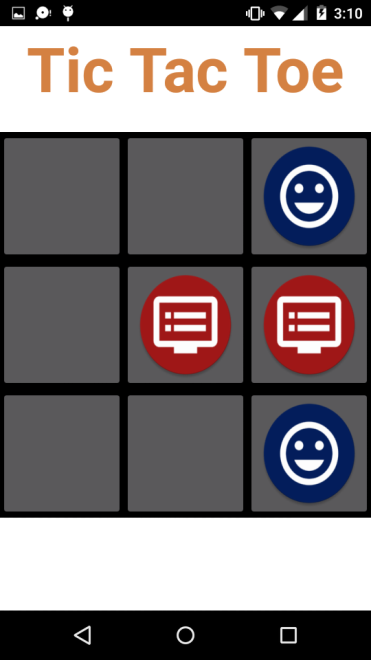
Easy– In this level it is very easy to win, even a person playing for the first time can win.

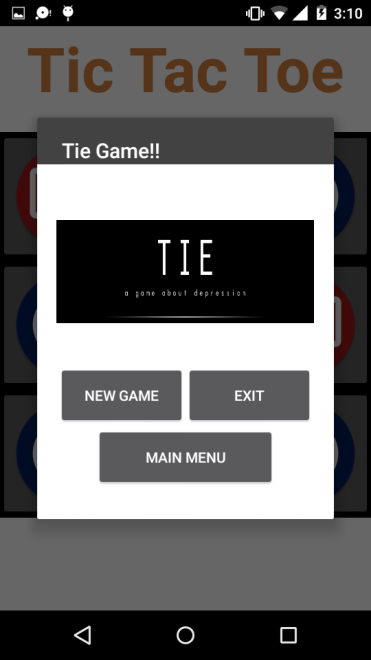
Hard - In this level it is very hard to win, unless you are master of this game.

Screenshot 6 Screenshot 7

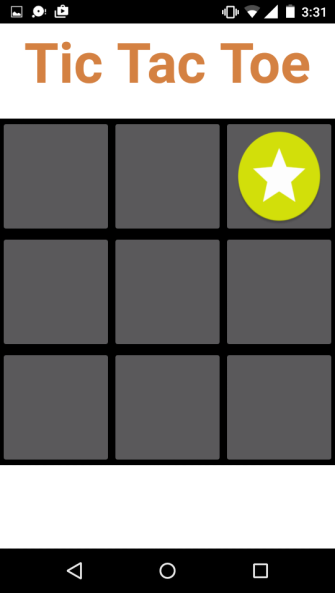
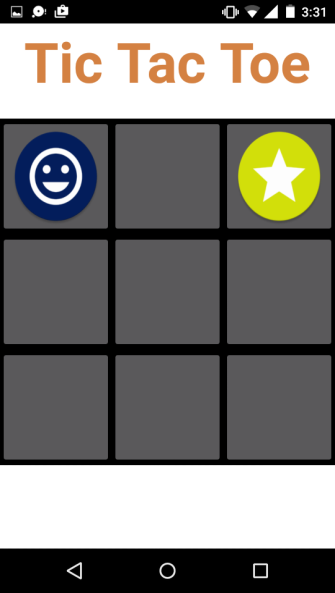
* + **NO. OF PLAYERS**
* **Playing against CPU**

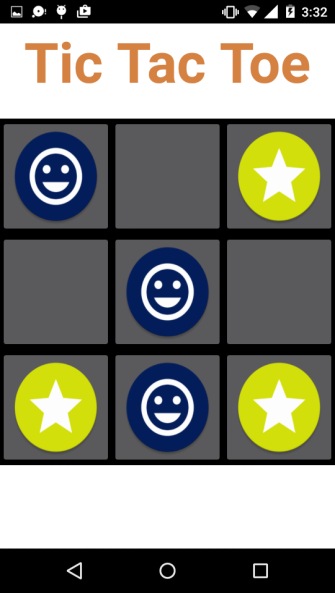
**  **

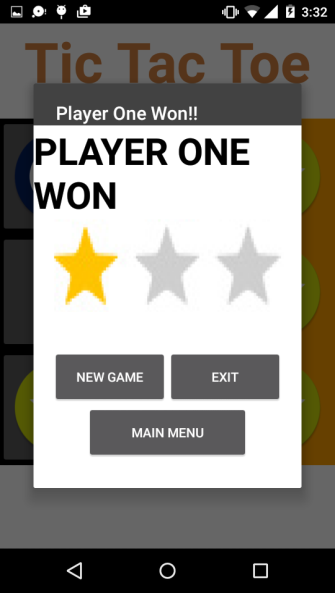
** **

Screenshot 8: Playing against CPU

* **Playing against human**

**  **

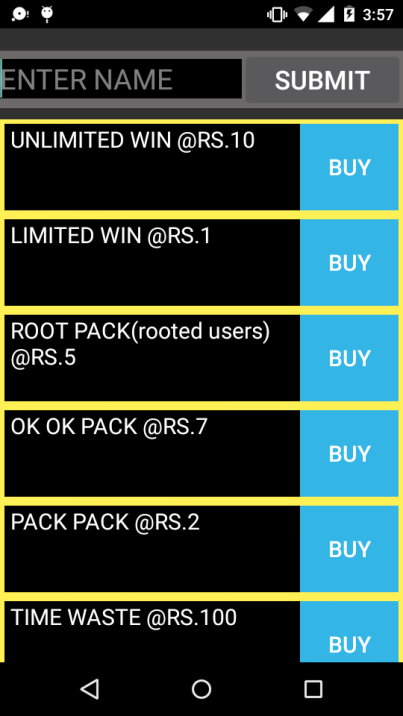
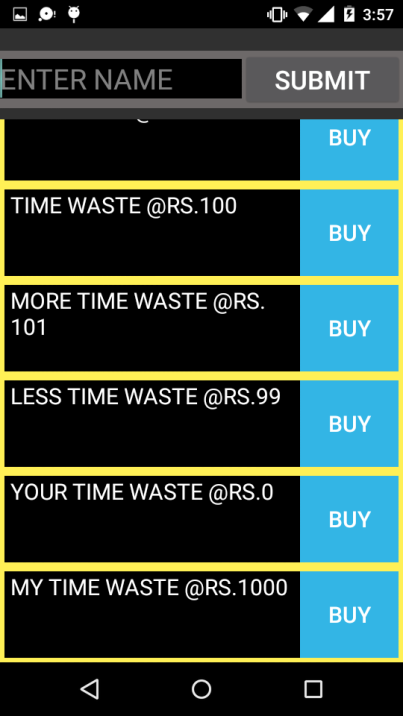
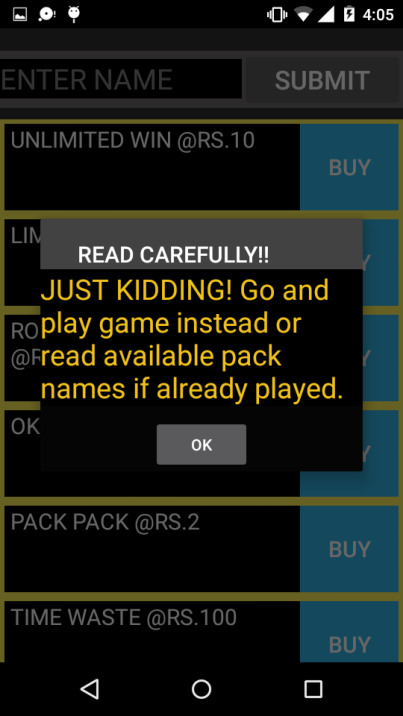
**  **

****

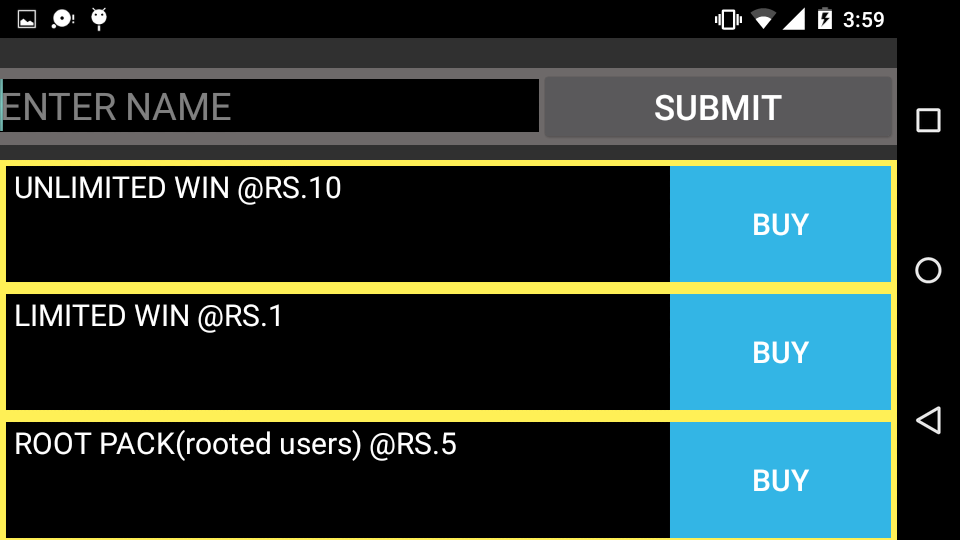
Screenshot 9: Playing against Human

* 1. **Buy Pro Pack**

When the user clicks on BUY PRO PACK option then it shows the list of Pro Packs:-

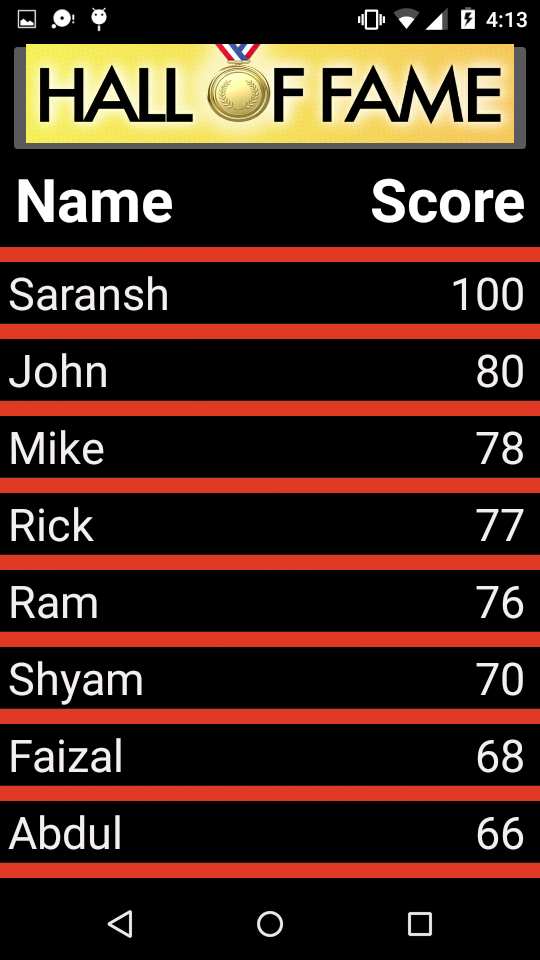
*Landscape View:*



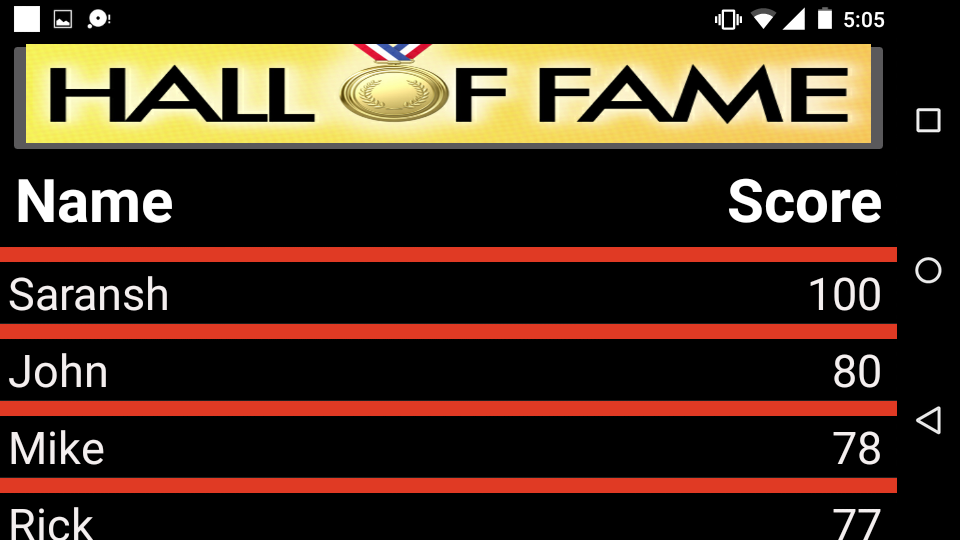
Screenshot 10: Buy Pro Pack

* 1. **High Score**

When the user clicks on High Score option then it shows the list of Scores:-



*Landscape View:*

**

Screenshot 11: High Scores

**7. Future Enhancements:**

In future, it can be made possible to:

* Providing Game Sounds: The game must also provide background sounds. The user also has the facility to mute the sounds whenever he/she pleases.
* Providing Instructions to play the Game: For players who are new to the game, instructions are given, so that they too can play.
* Providing Hints: Hints are to be given whenever the player requests hints.
* Adding Animations: Animations can be implements in the application which will improve the working and presentation of the application.
  + 1. **CONCLUSIONS**

Android is truly open, free development platform based on Linux and open source. Handset makers can use & customize the platform without paying a royalty. Android is open to all: industry, developers and users Participating in many of the successful open source projects Aims to be as easy to build for as the web Google Android is stepping into the next level of Mobile Internet.

We develop and implement tic-tac-toe game in android studio using the platform of the Android SDK. In this system, we create a 3x3 and 4x4 tic-tac-toe game in android. The system is designed so that a single or two players can play a game of tic-tac-toe using this application. The program will contain a display function and a select function to place the symbol as well as toggle between the symbols allowing each player a turn to play the game. The program will update after each player makes their move and will check for the conditions of the game as it goes on. Overall the system works without any bugs and is able to use.

In the end we would like to conclude that my aim to make this project was to research in the field of Android Application development and implementation by developing the logic for the game. Some scope of improvements is there in the project which will be rectified in the future advancements of the project. We would like to thank all those who have helped us and contributed in making of this project.

* + 1. **BIBLIOGRAPHY**

1. http://www.stackoverflow.com
2. http://www.developer.android.com
3. http://www.vogella.com
4. https://www.github.com
5. http://www.achartengine.org