COLLEGE OF ENGINEERING, THALASSERY

(Under CAPE, Estd. by Govt. Of Kerala)

KANNUR DT.-670107

MAJOR PROJECT REPORT ON

FAIRPLAY

Submitted in partial fulfilment of the requirements for the
Award of the degree of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE & ENGINEERING

By

LIJIN TV (Register No.:12152012)
VISHNU KV (Register No.:12152025)
NIMISHA VASANTHAKUMAR (Register No.:12152045)
SUNISHA C (12152054)
SWATHI K RANJITH (Register No.:12152056)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

MARCH 2018

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

COLLEGE OF ENGINEERING, THALASSERY

(Under CAPE, Estd. by Govt. Of Kerala)

KANNUR DT.-670107

CERTIFICATE

This is to certify that the major project work entitled

FAIRPLAY

Submitted by

LIJIN TV (Register No.:12152012)
VISHNU KV (Register No.:12152025)
NIMISHA VASANTHAKUMAR (Register No.:12152045)
SUNISHA C (12152054)
SWATHI K RANJITH (Register No.:12152056)

Is a bonafide record of the work done by them in partial fulfilment of the requirements for the award of B.Tech Degree in Computer Science during the year 2018.

PROJECT CO-ORDINATOR	**PROJECT GUIDE**	**HEAD OF THE DEPARTMENT**
Mrs. Priya V.V	Mrs. Priya V.V	Mrs. Ambili M P
Asst. Professor	Asst. Professor	Asst. Professor
Department of Computer	Department of Computer	Department of Computer
Science and Engineering	Science and Engineering	Science and Engineering

ACKNOWLEDGEMENT

We are greatly indebted to all those who have rendered help for successful completion of this project. First, we wish to extend our sincere gratitude to our principal**, Prof. Dr. Joseph O A.** for the help extended by him in providing all the facilities for our project. We have to extend our genuine and hearty thanks to **Mrs. Ambili**, Head, Department of Computer Science & Engineering for having permitted us to undergo this project. My sincere thanks to **Mrs. Priya V V** (Assistant Prof. in Computer Science), **Mrs. Bincy Antony** (Assistant Prof. in Computer Science), **College Of Engineering Thalassery** can’t be expressed fully in words. I thank her from the bottom of my heart. We also express our deepest thanks to our family members and friends for their moral support rendered to us to make this project a grant success. Above all we are greatly indebted to the Lord Almighty for giving us the courage and strength for the completion of this project work successfully.

ABSTRACT

Creation of an Android app "Fair play". The motivation behind developing such an app is due to the increasing number of malware attacks by online fraudsters. The app ensures tight security in Android platform and keeps the phone away from malwares and search rank frauds. The traditional app stores have a very weak detection rate in the case of such undesirable applications. The created app achieves more than 90% detection of such fraudulent applications and acts as a safe guard from attackers. The app developed not just detect such apps, but also gives the user an option to uninstall it and also provides suggestions showing better apps in the same category. The detection is done using the aspect of permissions, categories, and opinion mining of reviews provided by the user. This app is developed using Android studio.

To identify malware, previous work has focused on app executable and permission analysis. Ranking fraud in the mobile App market refers to fraudulent or deceptive activities which have a purpose of bumping up the Apps in the popularity list. Indeed, it becomes more and more frequent for App developers to use shady means, such as inflating their Apps’ sales or posting phony App ratings, to commit ranking fraud. While the importance of preventing ranking fraud has been widely recognized, there is limited understanding and research in this area. In this project, we introduce FairPlay, a novel system that discovers and leverages traces left behind by fraudsters, to detect both malware and apps subjected to search rank fraud. FairPlay correlates review activities and uniquely combines detected review relations with linguistic and behavioral signals gleaned from Google Play app data, in order to identify suspicious apps.

TABLE OF CONTENTS

CH. NO **TITLE** **PAGE NO**

LIST OF TABLES

1. INTRODUCTION

The commercial success of Android app markets such as Google Play and the incentive model they offer to popular apps, make them appealing targets for fraudulent and malicious behaviors. Some fraudulent developers deceptively boost the search rank and popularity of their apps (e.g., through fake reviews and bogus installation counts) while malicious developers use app markets as a launch pad for their malware. The motivation for such behaviors is impact: app popularity surges translate into financial benefits and advanced malware creation.

Fraudulent behaviour Google Play, the most popular Android app market, fuel search rank abuse and malware creation. To identify malware, previous work has focused on app executable and permission analysis. In this project introduce a novel system that discovers and controls traces left behind by fraudsters, to detect both malware and apps subjected to search rank fraud. This system correlates review activities and uniquely combines detected review relations with semantic and behavioral signals collected from Google Play app data in order to identify suspicious apps

In this project, we identify both malware and search rank fraud subjects in Google Play. This combination is not arbitrary: we posit that malicious developers resort to search rank fraud to boost the impact of their malware. Unlike existing solutions, we build this work on the observation that fraudulent and malicious behaviours leave behind telltale signs on app markets. We uncover these nefarious acts by picking out such trails. For instance, the high cost of setting up valid Google Play accounts forces fraudsters to reuse their accounts across review writing jobs, making them likely to review more apps in common than regular users. Resource constraints can compel fraudsters to post reviews within short time intervals. Legitimate users affected by malware may report unpleasant experiences in their reviews. Increases in the number of requested permissions from one version to the next, which we will call “permission ramps”, may indicate benign to malware transitions.

A valid user can login and search any app. For this an user first need to get registered to our app by entering the details like name, username, password, email id and date of birth. After registering the user can login and can search for any app. The related apps will also be listed by sorting the apps based on their rank. This avoids the rank fraud in Google Play. The apps will be listed on the basis of rank fraud, i.e., the least fraud app will be listed on top. Then the user can install the searched app and Fairplay will detect the its malicious behavior by counting and viewing permissions, rating and reviews. Once the installed app is detected as malware the user can uninstall if needed. The app has a provision to suggest the similar apps.

1.1 OBJECTIVE

To develop an android application that filters and sort out various application available on Google Play Store (Google App Store) on the basis of interests from various users and based on the number of downloads.

It also detects malware among the available apps and warns the user. In general, it is an android application used for choosing the right application for the particular use.

1.2 MOTIVATION

Increasing number of malware and of fraudulent apps in the app store has resulted in increasing rules in leakage of important or confidential data of the user. An android user is always prone to such attacks.

So a secure system is required in order to prevent such threats. The proposal system is really a great step towards android.

1.3 APPLICATION

- It is used to detect fraud ranking.
- It also used to detect malwares apps.

2. LITERATURE REVIEW

Yajin and Tiang *et al.* [2] proposed the Dissecting Android Malware Characterization and Evaluation.In this paper, the popularity and adoption of smart phones has greatly stimulated the spread of mobile malware, especially on the popular platforms such as Android. In light of their rapid growth, there is a pressing need to develop effective solutions. This paper focus on the Android platform and aim to systematize or characterize existing Android malware. Particularly, with more than onee yar effort, they managed to collect more than 1,200 malware samples that cover the majority of existing Android malware families. In addition, we systematically characterize them from various aspects, including their installation methods, activation mechanisms as well as the nature of carried malicious payloads. The characterization and a subsequent evolution-based study of representative families reveal that they are evolving rapidly to circumvent the detection from existing mobile anti-virus software. These results clearly call for the need to better develop next-generation anti-mobile-malware solutions.

In light of the threats, a system called AppCage that thoroughly confines the run-time behavior of third-party Android apps. It leverages two complimentary user-level sandboxes to interpose and regulate the app’s access to sensitive APIs, and further block malicious behaviors of Android malware. Specifically, the first sandbox named dex sandbox hooks into the app’s Dalvik virtual machine instance and redirects each sensitive framework API to a proxy which strictly enforces the user-defined policies, and the second sandbox named native sandbox leverages software fault isolation to prevent app’s native libraries from directly accessing the protected APIs or subverting the dex sandbox. Our evaluation showed that AppCage can successfully detect and block attempts to leak private information by third-party apps, and the performance overhead caused by AppCage is negligible for apps without native libraries and minor for apps with them.

IkerBurguera *et al.* [3] proposed the paper Behavior-Based Malware Detection System for Android.In this paper they capitalize on earlier approaches for dynamic analysis of application behavior as a means for detecting malware in the Android platform. The detector is embedded in a overall framework for collection of traces from an unlimited number of real users based on crowdsourcing. Our framework has been demonstrated by analyzing the data collected in the central server using two types of data sets: those from artificial malware created for test purposes, and those from real malware found in the wild. The method is shown to be an effective means of isolating the malware and alerting the users of a downloaded malware. This shows the potential for avoiding the spreading of a detected malware to a larger community.

As the results obtained with self-written malware in the system were successful, to make a deeper analysis for malware contained in Steamy Window and Monkey Jump 2 applications, using the Crowdroid client. Steamy Window application with PJApps malware Steamy Window is a free application available at the Android Market that covers the screen of the smartphone with steam and lets the user to wipe it off with the fingers. The malicious version of the application containing PJApps malware, which was discovered in unofficial repositories, sends sensitive information containing the IMEI, Device ID, Line Number and Subscriber ID to a web server. Then the infected smartphone gets registered in a Command and Control botnet waiting for instructions. It has the ability to send text messages to premium-rate numbers, SMS-spamming, install more applications, navigate, and even bookmark websites. Second experiment with real malware is done using a game called Monkey Jump 2. Even if this application is free and can be installed via Android Market, HongTouTou is included in repackaged apps made available through a variety of alternative app markets and forums targeting Chinesespeaking users. When Monkey Jump 2 infected with HongTouTou is executed, it sends device IMEI and IMSI data to a remote host. Then it receives instructions to click on web search result sites depending on received keywords. It also has the ability to download an application with the ability to monitor SMS conversations, and insert spam contents on them.

Asaf Shabtai *et al.* [4] proposed an article Behavioral Malware Detection Framework for Android Devices. This article presents Andromaly a framework for detecting malware on Android mobile devices. The proposed framework realizes a Host-based Malware Detection System that continuously monitors various features and events obtained from the mobile device and then applies Machine Learning anomaly detectors to classify the collected data as normal (benign) or abnormal (malicious). Since no malicious applications are yet available for Android, they developed four malicious applications, and evaluated Andromaly's ability to detect new malware based on samples of known malware.

Google's Android is a comprehensive software framework targeted towards such smart mobile devices (i.e., smartphones, PDAs), and it includes an operating system, a middleware and a set of key applications. Android emerged as an open-source, community-based framework which provides APIs to most of the software and hardware components. Specifically, it allows third-party developers to develop their own applications. The applications are written in the Java programming language based on the APIs provided by the Android Software Development Kit (SDK), but developers can also develop and modify kernel-based functionalities, which is not common for smartphone platforms. The security model of Android (and that of many other phone operating systems) is “system centric” (i.e., the system focuses on protecting itself). Applications statically identify the permissions that govern the rights to their data and interfaces at installation time. However, the application/developer has limited ability thereafter to govern to whom those rights. In order to overcome this limitation we propose a lightweight Malware Detection System (in terms of CPU, memory and battery consumption) for Android-based mobile devices in order to assist users in detecting (and optionally reporting to the Android community) suspicious activities on their handsets. The basis of the malware detection process consists of real-time, monitoring, collection, preprocessing and analysis of various system metrics, such as CPU consumption, number of sent packets through the Wi-Fi, number of running processes and battery level. System and usage parameters, changed as a result of specific events, may also be collected (e.g., keyboard/touchscreen pressing, application start-up). After collection and preprocessing, the system metrics are sent to analysis by various detection units, namely processors, each employing its own expertise to detect malicious behavior and generate a threat assessment (TA) accordingly.

Ezra Siegal [5] proposed an article Fake Reviews in Google Play and Apple App Store. Faking reviews and gaming the app stores seems to be a practice that is getting used more broadly. Both the Apple App Store and Google Play Store have made multiple changes to their ranking algorithms over the past year and some of the most recent shifts have clearly increased the importance of app ratings so the benefit of faking reviews is growing larger.

While it is impossible to always know if reviews were paid or not, there are some very clear examples of apps who have fake reviews. For example, by looking through reviews on a per reviewer basis, you can spot apps that are all using the same services to acquire app reviews and have similar reviews from the same group of people in the app store. Notice the similarity in brevity and phrasing.Another way to identify fake reviews is to look for review text that is identical – this happens more often than you might think. Fake app store reviews aren’t limited to just positive reviews either – we hear from clients all the time that one of the reasons they love having Apptentive on their side is to combat their competitors who are willing to pay for fake negative reviews. Similarly to the fake positive reviews, you don’t have to look very hard to find large numbers of 1 star reviews that use the exact same language to denigrate an app, hoping to drop it in the app store search results and rankings.

Zach Miners Report [6] proposed Malware-infected Android apps spike in the Google Play store. The number of mobile apps infected with malware in Google’s Play store nearly quadrupled between 2011 and 2013, a security group has reported.

In 2011, there were approximately 11,000 apps in Google’s mobile marketplace that contained malicious software capable of stealing people’s data and committing fraud, according to the results of a study published Wednesday by RiskIQ, an online security services company. By 2013, more than 42,000 apps in Google’s store contained spyware and information-stealing Trojan programs, researchers said.

Apps designed to personalize people’s Android-based phones were most susceptible, as well as entertainment and gaming apps. Some of the most malicious apps in the Google Play store downloaded since 2011 were Wallpaper Dragon Ball, a wallpaper app, and the games Finger Hockey and Subway Surfers Free Tips.

Both Wallpaper Dragon Ball and Finger Hockey, RiskIQ said, have malware that steals confidential information such as device IDs from infected devices. Subway Surfers Free Tips, meanwhile, uses a Trojan called Air Push to bypass a device’s security settings and subscribe infected phones to premium services.

RiskIQ performed its analysis using its own software that crawls app stores, websites and web ads. The technology, the company said, exposes malware that would otherwise not show itself to traditional web crawler software.

Android apps were only counted as being malicious if they behaved in specific ways as a result of malware. The behavior may include: collecting and sending GPS coordinates, contact lists and email addresses to third parties; recording phone conversations and sending them to attackers; taking control of the infected phone; or downloading other malware onto the phone.

Apps in Apple’s store were not analyzed. The findings show that the rising prominence of mobile apps among consumers also makes them a juicy target for hackers. Reports of possible malware in clones of the popular Flappy Bird mobile game recently surfaced, even after it was removed from app stores.

Malicious apps are an effective way to infect users, he said, since they often exploit the trust people have in brands and companies they do business with. But while the number of malicious Android apps is rising, the percentage of them removed by Google is on the decline, researchers said. In

2011 Google removed 60 percent of malicious apps, but in 2013 the company removed less than a quarter of them, the report said. That’s probably due to the rapid increase in malicious software. The overall number of malicious apps removed by Google still increased from roughly 7,000 in 2011 to nearly 10,000 in 2013.

Leman Akoglu *et al.* [7] proposed Opinion Fraud Detection in Online Reviews by Network Effects. User-generated online reviews can play a significant role in the success of retail products, hotels, restaurants, etc. However, review systems are often targeted by opinion spammers who seek to distort the perceived quality of a product by creating fraudulent reviews. A fast and effective framework, FRAUDEAGLE, for spotting fraudsters and fake reviews in online review datasets. Our method has several advantages: (1) it exploits the network effect among reviewers and products, unlike the vast majority of existing methods that focus on review text or behavioral analysis, (2) it consists of two complementary steps; scoring users and reviews for fraud detection, and grouping for visualization and sensemaking, (3) it operates in a completely unsupervised fashion requiring no labeled data, while still incorporating side information if available, and (4) it is scalable to large datasets as its run time grows linearly with network size. The effectiveness of our framework on synthetic and real datasets; where FRAUDEAGLE successfully reveals fraud-bots in a large online app review database.

Step 1: FRAUDEAGLE Scoring

Finding the best assignments to unobserved variables in our objective function, as given in Equation is the inference problem. In general, exact inference is known to be an NPhard problem, therefore we use a computationally tractable (in fact linearly scalable with network size) approximate inference algorithm called Loopy Belief Propagation (LBP). LBP is based on iterative message passing, and while it is provably correct only for certain cases, it has been shown to perform extremely well for a wide variety of applications in the real world. In the following we propose a new algorithm that extends LBP in order to handle signed networks. At convergence, we use the maximum likelihood label probabilities for scoring. signed Inference Algorithm (sIA)

Step 2. FRAUDEAGLE Grouping

Marginal class probabilities over users, products, and reviews, i.e. scores, enable us to order each set of them in a ranked list. While a rank list of, say, users with respect to being a fraudster is a valuable resource, it does not put the top such users in context with the products that they rated. In order to help with visualization, summarization, and further sensemaking of fraudsters we project the top users back on the review graph, and obtain the induced subgraph including these users along with the union of products that they rated. The idea is to partition this subgraph into clusters to gain more insight about how they are organized in the network. For partitioning, one can use any graph clustering algorithm. The cross-associations (CA) clustering algorithm (Chakrabarti et al. 2004) on the adjacency matrix of the induced graph. The CA algorithm performs clustering by finding a permutation of the rows (users) and columns (products) of the matrix such that the resulting matrix contains homogeneous blocks (defined by the clustering), where dense blocks correspond to near-bipartite cores (e.g., a team of users attacking on a target set of product).

Acar Tamersoy *et al* [2] proposed Guilt by association: Large scale malware detection by mining file-relationgraphs. The increasing sophistication of malicious software calls for new defensive techniques that are harder to evade, and are capable of protecting users against novel threats. We present Aesop, a scalable algorithm that identifies malicious executable files by applying Aesop's moral that "a man is known by the company he keeps". A large dataset voluntarily contributed by the members of Norton Community Watch, consisting of partial lists of the files that exist on their machines, to identify close relationships between files that often appear together on machines. Aesop leverages locality-sensitive hashing to measure the strength of these inter-file relationships to construct a graph, on which it performs large scale inference by propagating information from the labeled files (as benign or malicious) to the preponderance of unlabeled files. Aesop attained early labeling of 99% of benign files and 79% of malicious files, over a week before they are labeled by the state-of-the-art techniques, with a 0.9961 true positive rate at flagging malware, at 0.0001 false positive rate.

3. SYSTEM ANALYSIS

System analysis is the process of gathering and interpreting facts, diagnosing problems and using the information to recommend improvements on the system. It is a problem solving activity that requires intensive communication between system users and developers. System analysis or study is an important phase of any system development process. The system is viewed as a whole, the inputs are identified and the system is subjected to close study to identify the problematic areas. The solutions are given as a proposal. The proposal is reviewed on user request and suitable changes are made. This loop ends as soon as the user is satisfied with the proposal.

3.1 EXISTING SYSTEM

The existing site that provides a solution for this problem has many disadvantages like

- Malware detection systems are found to be inefficient.
- It is fewer users friendly.
- Identification and removal of malwares are not always successful.
- Google Play uses the Bouncer system to remove malware.
- Previous mobile malware detection work has focused on dynamic analysis of app executable as well as static analysis of code and permissions.
- It consider only malicious developers, who upload malware
- They do not ensure security.
- Percentage of accuracy is less.

3.2 PROPOSED SYSTEM

- This system identifies both malware and search rank fraud subjects in Google Play.
- This combination is not arbitrary: we suggest that malicious developers resort to search rank fraud to boost the impact of their malware.
- Unlike existing solutions, it builds this work on the observation that fraudulent and malicious behaviors leave behind telltale signs on app markets.
- It uncovers these immoral acts by picking out such trails. For instance, the high cost of setting up valid Google Play accounts forces fraudsters to reuse their accounts across review writing jobs, making them likely to review more apps in common than regular users.
- Resource constraints can compel fraudsters to post reviews within short time intervals.

- Legitimate users affected by malware may report unpleasant experiences in their reviews. Increases in the number of requested permissions from one version to the next, which will call “permission ramps”, may indicate benign to malware transitions.

3.3 FEASIBILITY STUDY

A feasibility study is a test of system proposal according to its workability, impact on the organization, ability to meet user’s needs and effective use of resources. The objective of feasibility study is acquiring a sense of the scope of the system and to avert a possible failure of the system after its creation. Three essential factors are involved in the feasibility analysis: Technical feasibility, Economic and Behavioral feasibility.

3.3.1 Economic Feasibility

Economic analysis is the most frequently used method for evaluating the effectiveness of the software, more commonly known as cost or benefit analysis. Here it is seen that no new hardware or software is needed for the development of the system. Hence, the project is economically feasible for development in this system.

3.3.2 Operational Feasibility

Project has been developed in such a way that it becomes very easy even for a person with little knowledge to operate it. This app is very user friendly and doesn’t require any technical skill to operate. Thus the project is even operationally feasible.

3.3.3Technical Feasibility

Technical feasibility centers on whether we can put the system in place using the technology at hand.

4. SYSTEM SPECIFICATION

4.1 SERVER SIDE

4.1.1 HARDWARE SPECIFICATION

The selection of hardware is very important in the existence and proper working of any software. Then selection hardware, the size and capacity requirements are also important.

Processor	:	Pentium Dual Core
Primary Memory	:	256MB RAM and above
Storage	:	40 GB hard disk and above
Display	:	VGA Colour Monitor
Key Board	:	Windows compatible
Mouse	:	Windows compatible

4.1.2 SOFTWARE SPECIFICATION

One of the most difficult task is selecting software for the system, once the system requirements is found out then we have to determine whether a particular software package fits for those system requirements. The application requirement:

Front end	:	JAVA, HTML, XHTML, XML, JAVA SCRIPT, CSS
Back end	:	MySQL ,JSP, Android Development Kit.
Operating system	:	windows 7 and above
IDE	:	Net beans

4.2 ANDROID

4.2.1 HARDWARE REQUIREMENTS

A mobile phone with **Android** operating system

Version	:	Android 4.1 or above
RAM	:	4 GB

4.2.2 SOFTWARE REQUIREMENTS

platform	:	WINDOWS//ANDROID
Front End	:	Java (JDK 6), XML (Android Development Tool)
IDE	:	android studio
Software used for development	:	Android Development Kit (Plug-in to the Eclipse IDE)

5. DEVELOPMENT TOOLS

5.1 Front End

The front end for server side is designed using Java and Android application using Java and XML

5.1.1 Java

Java is a general- purpose computer programming language that is concurrent, class-based, object oriented, and specifically designed to have as few implementation dependencies as possible. It is intended to let application developers "write once, run anywhere" (WORA), meaning that compiled Java code can run on all platforms that support Java without the need for recompilation. Java applications are typically compiled to byte code that can run on any Java virtual machine (JVM) regardless of computer architecture. As of 2015, Java is one of the most popular programming languages in use, particularly for client-server web applications, with a reported 9 million developers. Java was originally developed by James Gosling at Sun Microsystems (which has since been acquired by Oracle Corporation) and released in 1995 as a core component of Sun Microsystems' Java platform. The language derives much of its syntax from C and C++, but it has fewer low-level facilities than either of them.

Features of Java

- **Distributed**

  Java has an extensive library of routines for coping with TCP/IP protocols like HTTP and FTP Java applications can open and access across the Net via URLs with the same ease as when accessing local file system.

  We have found the networking capabilities of Java to be both strong and easy to use. Anyone who has tries to do Internet programming using another language will revel. How simple Java makes onerous tasks will like opening a socket connection.

- **Robust**

  Java is intended for writing programs that must be readable in a Variety ways. Java puts a lot of emphasis on early checking for possible problems, later dynamic checking, and eliminating situations that are error prone. The single biggest difference between Java has a pointer model that eliminates the possibility of overwriting memory and corrupting data. The Java compiler detects many problems that in other languages would only show up at runtime. As for the second point, anyone who has spent hours chasing a memory leak cost by a printer bug will be very happy with this feature of Java.

Java gives you the best of both worlds. You need not pointers for everyday constructs like string and arrays. You have the power of pointers if you need it, for example, for like lists. And you have always-complete safety, Since you can never access a bad pointer or make memory allocation errors.

- **Secure**

  Java is intended to be used in networked/distributed environment toward that end; a lot of emphasis been placed on security. Java enables the contraction of virus-free, temper-free systems. Here is a sample of what Java's security features are supposed to keep a Java programming from doing:

  1. Overrunning the runtime stack.
  2. Corrupting memory outside its own process space.
  3. Reading or writing local files when invoked through a security-conscious class loader like Web browser

Java has a major role in making our project dynamic and for establishing connection with database.
JAVA : Java(core), J2EE
Tools : Eclipse IDE, Net beans
Technologies : JSP, EJB
Frameworks : Struts, Turbine, Tapestry, Spring, Grails, Maverick

5.1.2 Java server page

Java Server Page (JSP) is a technology for controlling the content or appearance of Web pages through the use of servlets, small programs that are specified in the Web page and run on the Web server to modify the Web page before it is sent to the user who requested it. Sun Microsystems, the developer of Java, also refers to the JSP technology as the Servlet application program interface (API). JSP is comparable to Microsoft's Active Server Page (ASP) technology. Whereas a Java Server Page calls a Java program that is executed by the Web server, an Active Server Page contains a script that is interpreted by a script interpreter (such as VBScript or JScript) before the page is sent to the user.

Features of JSP

- **Extension to Servlet**

  JSP is Extension to Servlet, it have all the features of servlet and it have also implicit objects, predefined tags, expression language and Custom tags in JSP, that makes JSP easy to develop any application.

- **Powerful**

  These are internally Servlet, means consists byte code, so that all java features are applicable in case of jsp like robust, dynamic, secure, platform independent.

- **Portable**

  JSP tags will process and execute by the server side web container, So that these are browser independent and j2ee server independent.

- **Flexible**

  Allows to defined custom tags, the developer can fill conferrable to use any kind, framework based markup tags in JSP.

- **Easy**

  JSP is easy to learn, easy to understand and easy to develop. JSPs are more convenient to write than Servlets because they allow you to embed Java code directly into your HTML pages, in case of servlets you embed HTML inside of Java code.

- **Less code than Servlet**

  In JSP, we can use a lot of tags such as action tags, jstl, custom tags etc. that reduces the code.

5.1.3 HTML (Hyper Text Markup Language)

HTML is a syntax used to format a text document on the web. These documents are interpreted by web browsers such as Internet Explorer and Netscape Navigator. HTML can be created as standard ASCII text with "tags" included to pass on extra information about character formatting and page layout to a web browser. HTML is written in the form of HTML elements consisting of tags enclosed in angle brackets (like <html>), within the web page content. HTML tags most commonly come in pairs like <h1> and </h1>, although some tags represent empty elements and so are unpaired, for example <img>. The first tag in a pair is the start tag, and the second tag is the end tag (they are also called opening tags and closing tags). In between these tags web designers can add text, further tags, comments and other types of text-based content. The purpose of a web browser is to read HTML documents and compose them into visible or audible web pages. The browser does not display the HTML tags, but uses the tags to interpret the content of the page. HTML elements form the building blocks of all websites. HTML allows images and objects to be embedded.

HTML has a major role in our project for web page creation.

5.1.4 CSS (Cascading Style Sheets)

CSS is a style sheet language used for describing the look and formatting of a document written in a markup language. While most often used to style web pages and interfaces written in HTML and XHTML, the language can be applied to any kind of XML document, including plain XML, SVG and XUL. CSS is a cornerstone specification of the web and almost all web pages use CSS style sheets to describe their presentation. CSS is designed primarily to enable the separation of document content from document presentation, including elements such as the layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple pages to share formatting, and reduce complexity and repetition in the structural content (such as by allowing for table less web design). CSS can also allow the same markup page to be presented in different styles for different rendering methods, such as on-screen, in print, by voice (when read out by a speech-based browser or screen reader) and on Braille-based, tactile devices. It can also be used to allow the web page to display differently depending on the screen size or device on which it is being viewed.

5.2 Back End

5.2.1 MySQL

MySQL is the world‟s second most widely used open-source relational database management system (RDBMS). The SQL phrase stands for Structured Query Language. The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation.

MySQL is a popular choice of database for use in web applications and is a central component of the widely used LAMP (Linux, Apache, MySQL, Perl/PHP/Python) open source web application software stack. Free-software-open source projects that require a full featured database management system often use MySQL.

For commercial use, several paid editions are available, and offer additional functionality. Applications which use MySQL database include: TYPO3, MODx, joomla, WordPress, phpBB, Drupal and other software. MySQL is also used in many high-profile, large-scale websites, including Wikipedia, Google (though not for searches), Facebook, Twitter, Flickr and YouTube.

Features of MySQL

The following list shows the most important properties of MySQL

- **Relational database system**

  Like most all other database systems on the market, MySQL is relational database system

- **Client/Server architecture**

  MySQL is client/Server system. There is a database server and arbitrarily many clients, who communicate with the server; that is, they query data, save changes etc. the clients can run on the same computer as the server or on another computer. Almost all of the familiar large database system such as Oracle ,Microsoft SQL, Server etc are client/server systems.

- **SQL Compatibility**

  MySQL supports as its database language as its name suggests-SQL.SQL is a standardized language for querying and updating data and for the administrator of the database. There are several SQL dialects. MySQL adheres to the current SQL standard although with significant restrictions and large number of extensions.

- **Scalability and limits**

  Support for large databases. We use MySQL server with databases that contain 50 million records.. Each index may consist of 1 to 16 columns or parts of columns. An index may use a prefix of a column for CHAR, VARCHAR, BLOB or TEXT column types.

- **Clients and tools**

  The MySQL server has a built-in support for SQL statements to check, optimize and repair tables. These statements are available from the command line through the mysqlcheck client. MySQL also includes myisamchk, a very fast command-line utility for performing these operations on MyISAM tables. All MySQL program can be invoked with the –help or-? Options to obtain online assistance.

- **Security**

  A privilege and password system that is very flexible and secure that allows host based verification. Passwords are secure because all password traffic is encrypted when you connect to a server.

- **Web and data warehouse strengths**

MySQL is the de-facto standards for high traffic websites because of its high performance query engines, tremendously fast data inserting capability and strong support for specialized web functions like fast full text searches. Other features like main memory tables, b-tree and hash indexes and compressed archive table that reduce storage requirements by to 80% make MySQL a strong standout for both web and business intelligence applications.

- **Robust transactional support**

  MySQL offers one of the most powerful transactional database engines on the market. Features include complete ACID (atomic, consistent, isolated, durable) transactional support, unlimited row-level locking, distributed transaction capability, and multi version transaction support where readers never block writers and vice versa.

5.3 Net Beans

NetBeans is a software development platform written in Java. The NetBeans Platform allows applications to be developed from a set of modular software components called modules. Applications based on the NetBeans Platform, including the NetBeans integrated development environment (IDE), can be extended by third party developers. The NetBeans IDE is primarily intended for development in Java, but also supports other languages, in particular PHP, C/C++ and HTML5. NetBeans is cross-platform and runs on Microsoft Windows, Mac OS X, Linux, Solaris and other platforms supporting a compatible JVM. The NetBeans Team actively support the product and seek feature suggestions from the wider community. The NetBeans Platform is a framework for simplifying the development of Java Swing desktop applications. The NetBeans IDE bundle for Java SE contains what is needed to start developing NetBeans plugins and NetBeans Platform based applications; no additional SDK is required. Applications can install modules dynamically. Any application can include the Update Center module to allow users of the application to download digitally signed upgrades and new features directly into the running application. Reinstalling an upgrade or a new release does not force users to download the entire application again.

Features of NetBeans IDE

- **Best support for latest Java Technologies**

  NetBeans IDE is the official IDE for Java8. With its editors, code analyzers, and converters, you can quickly and smoothly upgrade your applications to use new Java8 language constructs, such as lambdas, functional operations, and method references. Batch analyzers and converters are provided to search through multiple applications at the same time, matching patterns for conversion to new Java8 language constructs. With its constantly improving Java Editor, many rich features and an extensive range of tools, templates and samples, NetBeans IDE sets the standard for developing with cutting edge technologies out of the box.

- **Fast and Smart Code Editing**

  An IDE is much more than a text editor. The NetBeans Editor in dents lines, matches words and brackets, and highlights source code syntactically and semantically. It lets you easily refactor code, with arrange of handy and powerful tools, while it also provides code templates, coding tips, and code generators. The editor supports many languages from Java, C/C++, XML and HTML, to PHP, Groovy, Java doc, Java Script and JSP. Because the editor is extensible, you can plugin support for many other languages.

- **Easy and Efficient Project Management**

  Keeping a clear overview of large applications, with thousands of folders and files, and millions of lines of code, is a daunting task. NetBeans IDE provides different views of your data, from multiple project windows to helpful tools for setting up your applications and managing them efficiently, letting you drill down into your data quickly and easily, while giving you versioning tools via Subversion, Mercurial, and Git integration out of the box. When new developers join your project, they can understand the structure of your application because your code is well-organized.

- **Rapid User Interface Development**

  Design GUIs for Java SE, HTML5, Java EE, PHP, C/C++, and Java ME applications quickly and smoothly by using editors and drag-and-drop tools in the IDE. For Java SE applications, the NetBeans GUI Builder automatically takes care of correct spacing and alignment, while supporting in-place editing, as well. The GUI builder is so easy to use and intuitive that it has been used to prototype GUI slive at customer presentations.

- **Write Bug Free Code**

  The cost of buggy code increases the longer it remains unfixed. NetBeans provides static analysis tools, especially integration with the widely used Find Bugs tool, for identifying and fixing common problems in Java code. In addition, the NetBeans Debugger lets you place break points in your source code, add field watches ,step through your code, run into methods, take snapshots and monitor execution as it occurs.

The NetBeans Profiler provides expert assistance for optimizing your application's speed and memory usage, and make site easier to build reliable and scalable Java SE, Java FX and Java EE applications. NetBeans IDE includes a visual debugger for Java SE applications, letting you debug user interfaces without looking into source code. Take GUI snapshots of your applications and click on user interface elements to jump back into the related source code.

Due to the above advantages we chose NetBeans IDE over Eclipse IDE.

5.4 ANDROID STUDIO

Android Studio is an IDE based on IntelliJ IDEA used for android application development. This tool has more options for Android Development, making the process faster and more productive. Prior to Android Studio, developers were relying only on the open source eclipse as IDE with ADT plugin for android development. Due to this android was always falling back compared Apples xCode IDE for iOS based development. After android studio release Google can equally bet with iOS platform in terms of development assets. Now let’s see more of the IDE capabilities.

It is the best available platform for creating android apps. Hence we chose Android Studio.

Android Studio Features

Android studio is based on IntelliJ IDEA, which does all the functionality that Eclipse with ADT plug-in do, with lot more additional features. The initial version of android studio offers

1. Gradle-based build support.
2. Android-specific refactoring and quick fixes
3. Lint tools to catch performance, usability, version compatibility and other problems
4. ProGuard and app-signing capabilities
5. Template-based wizards to create common Android designs and components.

A rich layout editor: it allows you to drag-and-drop UI components, preview layouts on multiple screen configurations. Preview appears instantly as you change in the layout editor. You can choose a language, and can see the preview of layout with that locale.

Rich Color Preview editor: While adding colors as a resource, and we can see the color preview at the left hand side of the editor.

Deep Code Analysis: If you point to a line and it gives detailed explanation about an exception based on the annotation added. And you can also know which constants are allowed for which API.

It also has the powerful code completion. You can also inspect code in whole project, InteliJ lists all Lint errors during code inspection.

5.5 DREAMWEAVER

Dreamweaver Software, which is made by Adobe, is an award winning program that allows you to create websites and applications. With Dreamweaver, you can design visually or you can design by using code. Dreamweaver is compatible with both Windows and Mac operating systems.

It is mainly used in our project for creating HTML pages.

Here are three great features of this program:

1. **Integrated CMS Support**
   One of the best feature of Dreamweaver is its integrated CMS support. Dreamweaver allows you to author and test all of the popular content management systems, including Drupal, WordPress. The integrated CMS support in Dreamweaver even comes with live view navigation that allows you to see your webpage in action for easy editing. If you are working with dynamic pages, the integrated CMS support that is featured in Dreamweaver allows you to access any and all page related files.
2. **Intelligent Coding Assistance**
   Dreamweaver makes it easy for you to write clean code. If you are not too familiar with coding by hand, you can take advantage of the JavaScript, HTML and Ajax code hinting that is provided by Dreamweaver. Frameworks that Dreamweaver provides code hinting for include Prototype, Spry, and jQuery. Dreamweaver even comes with PHP code hinting that allows you to learn about popular PHP methods and functions.
3. **Comprehensive CSS Support**
   CSS tools are the new trend in website development, although they can be difficult to work with. Dreamweaver allows you to display the CSS box model without needing to know how to code CSS manually. Unlike other website creation programs, Dreamweaver does not require you to run separate utilities to create websites that come with powerful CSS tools.

5.6 Cost estimation

Following the COCOMO model for cost and effort estimation we find that ,

5.6.1 Effort estimation:

Effort = 3.0 × KLOC ˆ1.12 PM

For our project we estimate the Kilo Line Of Code(KLOC) as,12. On substituting the value of KLOC,

Effort = 3.0 × 10 ˆ 1.12 = 39.55PM

where, PM is the person month

5.6.2 Time estimation:

We estimate development time ,Tdev as,

Tdev = 2.5 × (Effort) ˆ 0.35 months

We obtained Effort = 4

8.51 PM Now, Tdev = 2.5 ×(39.55)ˆ0.35 = 9.05 months

Roughly calculating,

Cost for requirement analysis = 1000 /-

Cost for design = 36500/-

Cost for coding = 18,750 /-

Cost for testing = 18,750/-

Software Expense = 5,000/-

Hardware Expense = 20,000/-

Therefore, total cost for the project = 1,00,000/-

6. MODULAR DIVISION

Modules are divided into 3:

- **User**
- **Admin**
- **Services**

USER

1. REGISTRATION

After successful installation of FairPlay, the user can create an account by providing name, dob, email id, password, phone number and IMEI number is also extracted for one-time registration.

2. LOGIN

A valid user can login to our app by providing correct username and password.

3. SEARCH AND INSTALL

After login, the user can search for the required app and install them. The apps will be listed in the increasing order of fraud.

4. RATING AND REVIEW

The user can rate and can make review on our app.

5. VIEW SUGGESTION AND UNINSTALL

If the installed app is detected to be malware, the user can uninstall them. Suggestions about the related apps are also given.

ADMIN

1. LOGIN

Admin can login by providing name, password, email and phone number.

2. SET CATEGORY

Available categories will be already listed and new categories can be added by giving category name and description. Permissions corresponding to the category name can be also provided.

3. EDIT CATEGORY

Admin can modify category. That is, he/she can modify or delete category.

4. SET PERMISSION

Admin can add extra permission to the available category.

5. EDIT PERMISSION

Admin can modify or delete permission.

6. VIEW RATING AND REVIEW

The ratings and reviews given by the user are viewed by the admin.

SERVICES

1. RANK FRAUD DETECTION

Here, the user can search the required app without getting bothered about the search rank fraud. This app sorts the suggested application in the ascending order of fraud. Thus apps that are safe from fraud ranking will be sorted at the top.

The main operation under Search Rank Fraud module:

- Comparing the count of each reviewed user and then taking the overall percentage.

2. DETECTING MALWARE BY COMPARISION

We compare the app's permission with the permission of apps that falls to the same category to see whether it access any additional permissions. If the installed app is requesting extra permission, then we examine rating and reviews as the next step for detecting malware. If not, then the app is confirmed to be good.

We extract the rating of the installed app. Then the rating is compared with the threshold value. If the value is less than the threshold value, then it means the app has more chance to become malware.

Once the permission and rating gives a negative output, we analyse positive and negative reviews in details and then calculate the percentage of positive reviews and negative reviews. If the negative rating is greater than the positive reviews, the app is said to malware.

7. SYSTEM DESIGN

System design is the process of developing specifications for a candidate system that meets the criteria established in the system analysis. The major steps in system design are the preparation of the input and the output reports in a form applicable to the user. The main objective of the system design is to use the package easily by any computer operator. System design is the creative act of invention, developing new inputs, a database, offline files, methods, procedure and output for processing business to meet an organization objective. System design builds information gathered during the system analysis.

7.1 Input Design

Input design plays a vital role in the life cycle of software development. It requires very careful attention of developers. It specifies the manner in which the data enters the system for processing. Input design can ensure the reliability of the system and produces results from accurate data or it may result in output of enormous data. According to software Engineering concepts, the input forms of screens are designed to provide to have validation control over the input limit, range and other related validations.

This system has input screens in almost all the modules. Error messages have been developed to alert the user whenever he commits some mistakes and guides him in the right way. Input design determines whether the user interacts with the system efficiently. The link ties the information system into the user world. It consists of developing specifications and procedure for data preparation. Therefore, structured steps are necessary to put transaction data into usable form for processing.

Input design is the process of converting the user originated inputs to computer based formats. It also includes the method of input and entry into the system. There can be many factors that can be taken into account.

- Flexibility
- Speed
- Accuracy
- Type of input
- Verification methods
- Ease of use

Keyboard is the most commonly used input media. Inaccurate input data are the most common cause of errors in data processing. Error entered by the user the user can be controlled by the input design.

The design for handling input specifies how the data are accepted for computer processing. Input design is a part of overall system design that needs careful attention and it includes specifying the means by which action are taken.

A system user interacting through a workstation must be able to tell the system whether to accept the input produce a report or end the processing. The collection of input data is considered as the most extensive part of system design. Since the inputs have to be planned in such a manner to get the relevant information, extreme care has to be taken to obtain the information. If the data going into the system is incorrect, then the processing and the output ill magnify these errors.

Major activities carried out are

- Collection of needed data from the source
- Conversion of data into computer accepted form.
- Verification of converted data
- Checking the data for accuracy

Information is contained in the architectural and detailed specification. Functional and performance test are designed during design phase.

Software has three types of input:

- Admin side input
- User side input

Administrator is able to

- View the registered users
- Can verify users
- Manage database
- View the feedback given by registered and unregistered users
- Check for malware and search rank frauds
- Eliminate malware and search rank fraud

User is able to

- Register an account
- Search for any apps
- Install an app
- Edit profile
- Uninstall the app if needed

7.2 Output design

Often the most important system for users is the output it produces. Without quality, outputs we may even feel the entire system is so unnecessary that they avoid using it. The term output is applied to any information produced by the system whether printed, displayed or spoken.

Output design refers to the generation of results and information of the project. It should be an organized and well out manner. It involves conceiving, planning and specifying the external observable characteristics of the product. It includes user displays and report formats.

The normal procedure is to design the outputs in detail first and then to work back with the input. The outputs can be in the form of operational documents and reports. The input record have to be validated, edited, organized and accepted by the system before being processed to produce the output.

Output is the most important and direct source of information to the users. Designing the computer output should produce in an organized manner. The right output must be developed while ensuring that each output element is designed so that people will find the system user friendly.

The output design has to be done so that the results of processing should be communicated to the user. Effective output design will improve the clarity and performance of output. Output is the main reason for developing the system and the basis on which the usefulness of application will be evaluated. Output design phase of the system is concerned with the convergence of information to the end user in a user friendly manner. The output design should be efficient, intelligible so that system relationship with the end user improves and thereby enhancing the process of decision making.

7.3 Data Flow Diagram

The DFD is used for expressing system requirements in a graphical form. The DFD is also known as bubble chart, which is used for the purpose of clarifying system requirement and identifying major transformation that will become programs in system design. So this is the starting point of the design phase that functionally decomposes the requirement specification down to the lowest level of detail. A DFD consist of a series bubbles joined lines. The bubbles represent data transformation and the lines represent data flow in the system. In the normal convention, logical DFD can be completed using only 5 notations.

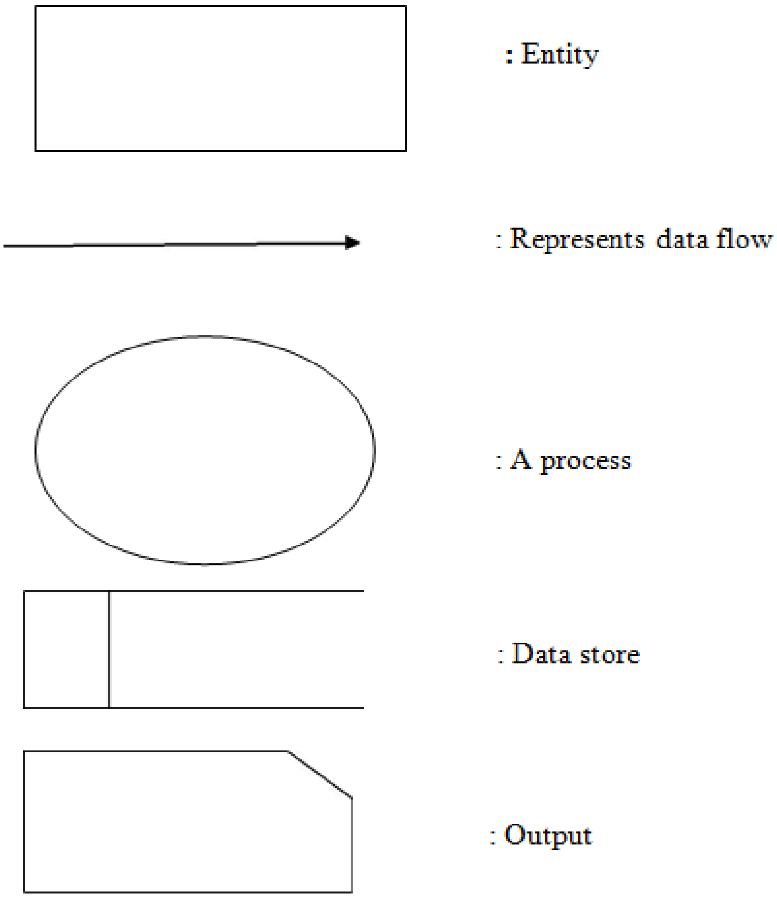
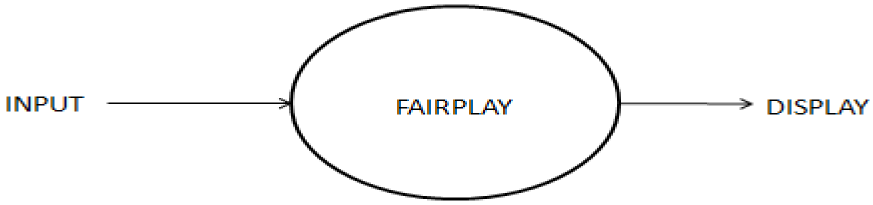
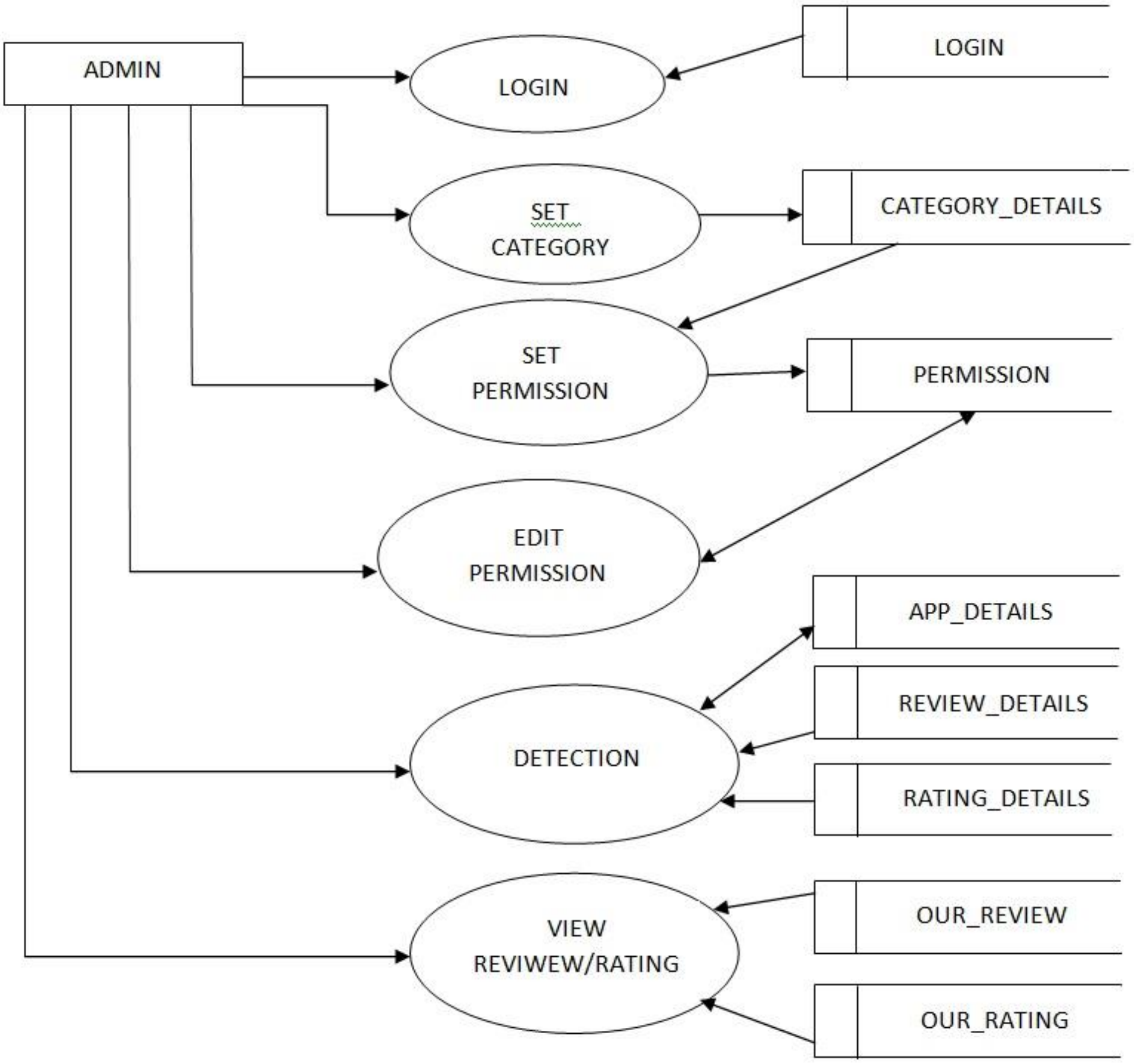


Figure 7.1 Basic symbols in DFD

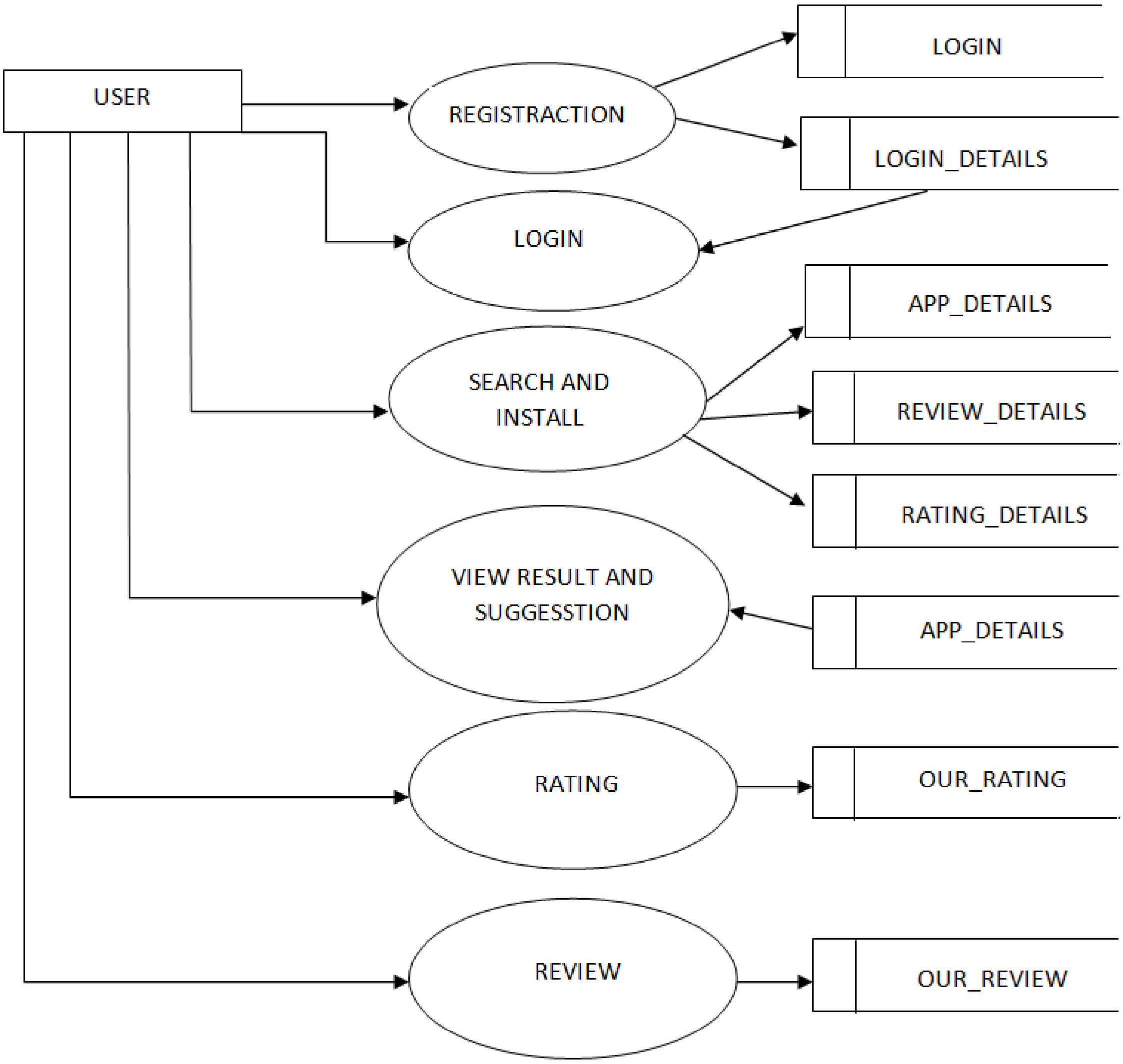
LEVEL 0



LEVEL 1.1



LEVEL 1.2


USER
REGISTRACTION
LOGIN
LOGIN_DETAILS
LOGIN
APP_DETAILS
SEARCH AND INSTALL
REVIEW_DETAILS
RATING_DETAILS
VIEW RESULT AND SUGGESSTION
APP_DETAILS
RATING
OUR_RATING
REVIEW
OUR_REVIEW

7.4 Database Design

Design process

Database design is required to manage the large bodies of information. The management of data involves both the definition of structure of information and provision of mechanism for the manipulation of information. In addition the database system must provide for the safety of information handled, despite the system crashes due to attempts of unauthorized access. For developing an efficient database, we will have to fulfil certain conditions such as

- Control redundancy
- Ease of use
- Data independence
- Accuracy and integrity
- Avoiding inordinate delays
- Recovery from failure
- Privacy and security
- Performance

There are 6 major steps in design process. The first 5 are usually done on paper and finally the design is implemented.

- Identify the tables and relationships
- Identify the data that is needed for each table and relationship
- Resolve relationship
- Verify relationship
- Implement the design

Normalization

Normalization is the process of analyzing the given relation schemes based on their functional dependencies and primary keys to achieve the desirable properties of

- Minimizing redundancy
- Minimizing the insertion, deletion and updating anomalies Normalization is carried out for the following reasons
- To structure the data so that perfect relationship between entries can be represented
- To permit simple retrieval of data in response query and report request
- To reduce the need to restructure or reorganize data when new application requirement

arises

Normalization consists of various levels:

- First Normal Form (1NF) :

A table is in 1NF if there are no duplicate rows in the table. Each cell is a single valued. Entries in a column are of the same kind

- Second normal form (2NF) :

Second normal for is based on the concept of full functional dependency. A table is in 2NF if it is in 1NF and if all non key attributes are dependent on a key.

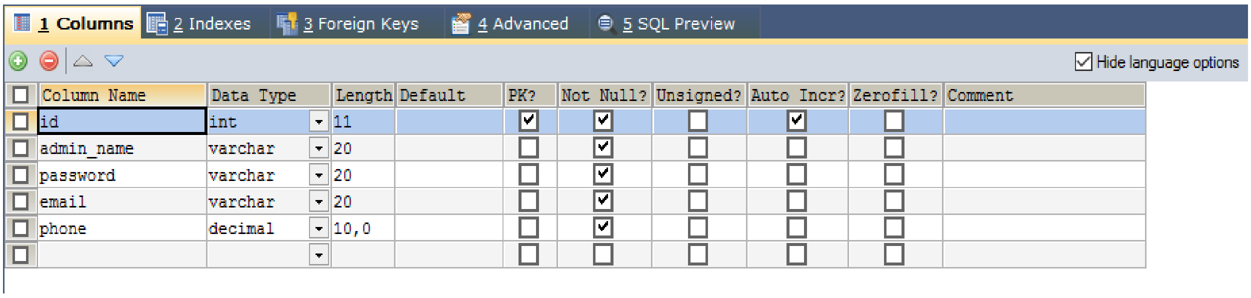
Dependent on only a part of the key, the definition of 2NF is sometimes phrased as “A table is in 2NF if it is in 1NF and it has no partial dependencies”.

- Third Normal Form (3NF) :

Third normal form is based on the concept of transitive dependency. A table is in 3NF if it is in 2NF and if it has no transitive dependencies.

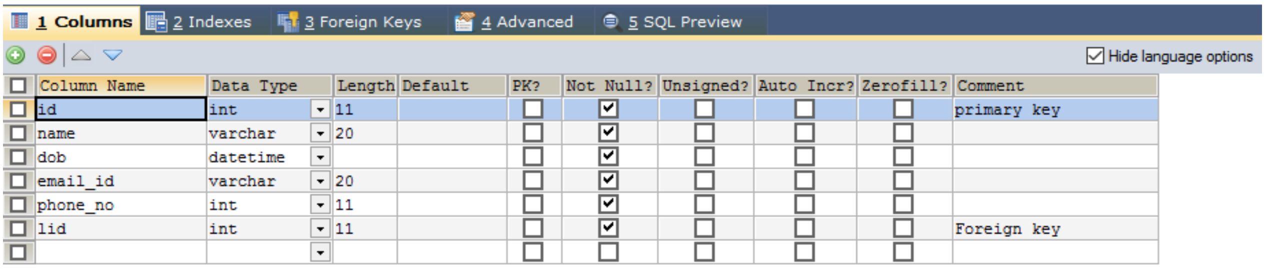
7. TABLES

7.1 admin_details



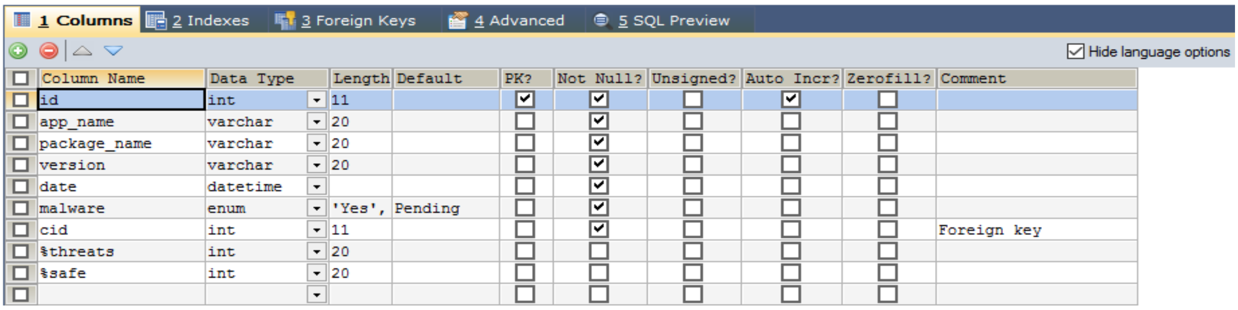
□	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	Comment
□	id	int	11		☑	☑	□	☑	□	
□	admin_name	varchar	20		□	☑	□	□	□	
□	password	varchar	20		□	☑	□	□	□	
□	email	varchar	20		□	☑	□	□	□	
□	phone	decimal	10,0		□	☑	□	□	□	
□					□	□	□	□	□	

7.2 user_details



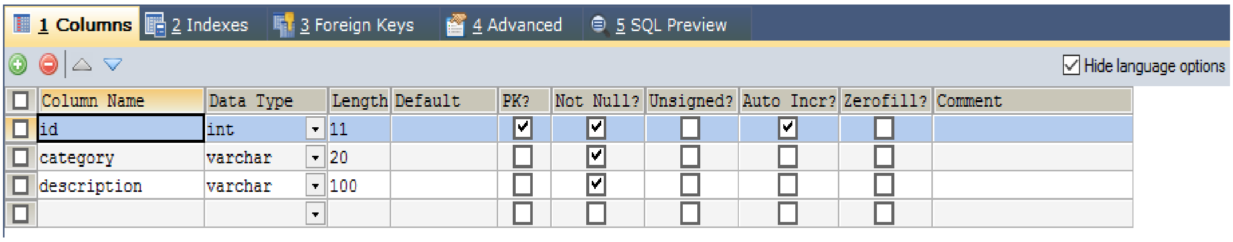
□	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	Comment
□	id	int	11		□	☑	□	□	□	primary key
□	name	varchar	20		□	☑	□	□	□	
□	dob	datetime			□	☑	□	□	□	
□	email_id	varchar	20		□	☑	□	□	□	
□	phone_no	int	11		□	☑	□	□	□	
□	lid	int	11		□	☑	□	□	□	Foreign key
□					□	□	□	□	□	

7.3 app_details



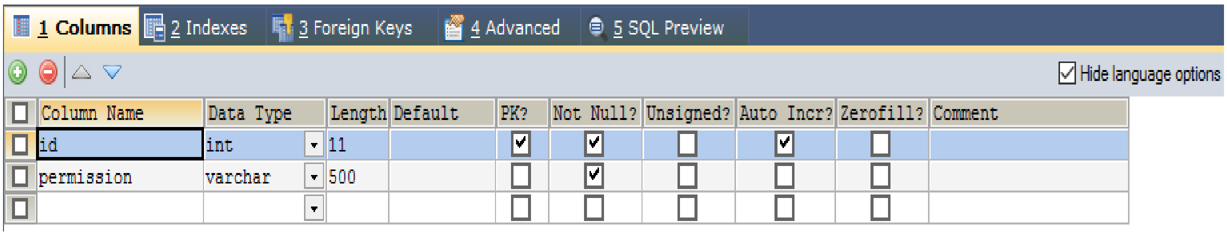
□	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	Comment
□	id	int	11		☑	☑	□	☑	□	
□	app_name	varchar	20		□	☑	□	□	□	
□	package_name	varchar	20		□	☑	□	□	□	
□	version	varchar	20		□	☑	□	□	□	
□	date	datetime			□	☑	□	□	□	
□	malware	enum	'Yes',	Pending	□	☑	□	□	□	
□	cid	int	11		□	☑	□	□	□	Foreign key
□	%threats	int	20		□	□	□	□	□	
□	%safe	int	20		□	□	□	□	□	
□					□	□	□	□	□	

7.4 category_details



□	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	Comment
□	id	int	11		☑	☑	□	☑	□	
□	category	varchar	20		□	☑	□	□	□	
□	description	varchar	100		□	☑	□	□	□	
□					□	□	□	□	□	

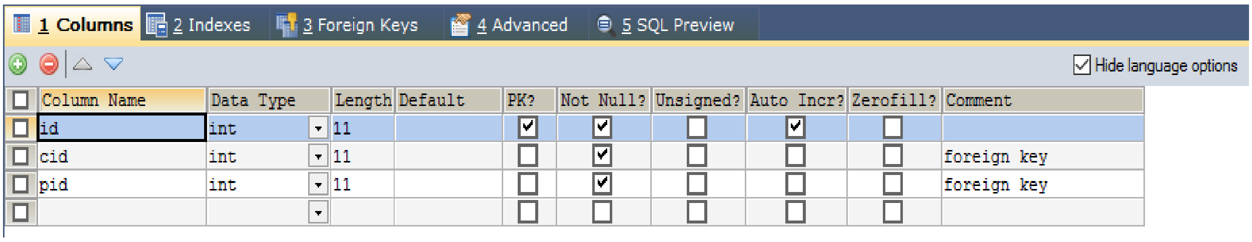
7.5 permission_details



☑ Hide language options

□	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	Comment
□	id	int	11		☑	☑	□	☑	□	
□	permission	varchar	500		□	☑	□	□	□	
□					□	□	□	□	□	

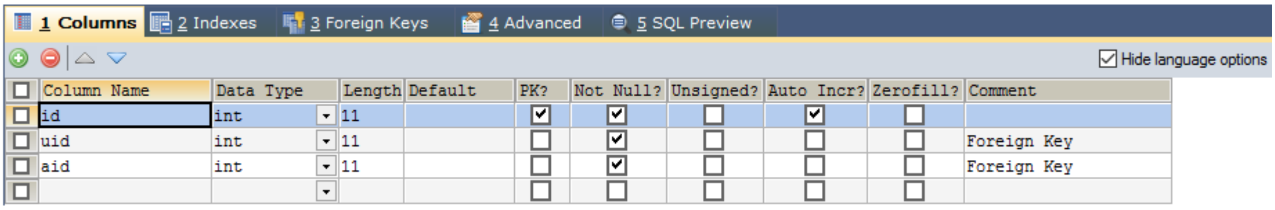
7.6 category_permission



☑ Hide language options

□	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	Comment
□	id	int	11		☑	☑	□	☑	□	
□	cid	int	11		□	☑	□	□	□	foreign key
□	pid	int	11		□	☑	□	□	□	foreign key
□					□	□	□	□	□	

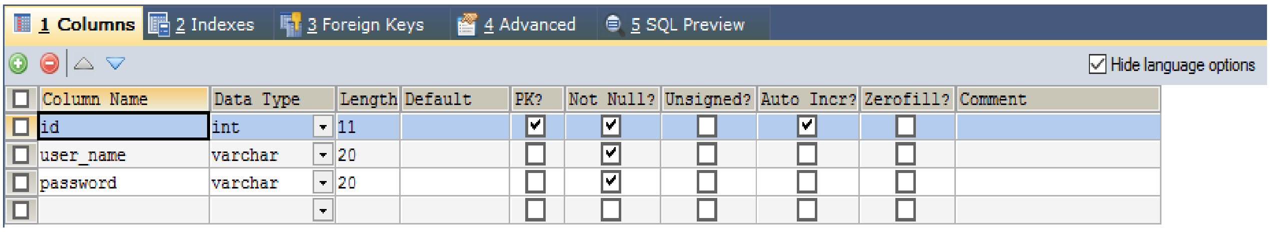
7.7 user_app



☑ Hide language options

□	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	Comment
□	id	int	11		☑	☑	□	☑	□	
□	uid	int	11		□	☑	□	□	□	Foreign Key
□	aid	int	11		□	☑	□	□	□	Foreign Key
□					□	□	□	□	□	

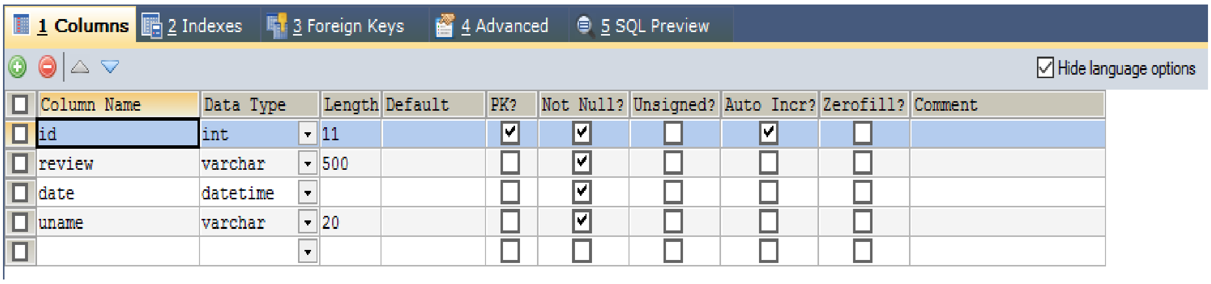
7.8login_details



☑ Hide language options

□	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	Comment
□	id	int	11		☑	☑	□	☑	□	
□	user_name	varchar	20		□	☑	□	□	□	
□	password	varchar	20		□	☑	□	□	□	
□					□	□	□	□	□	

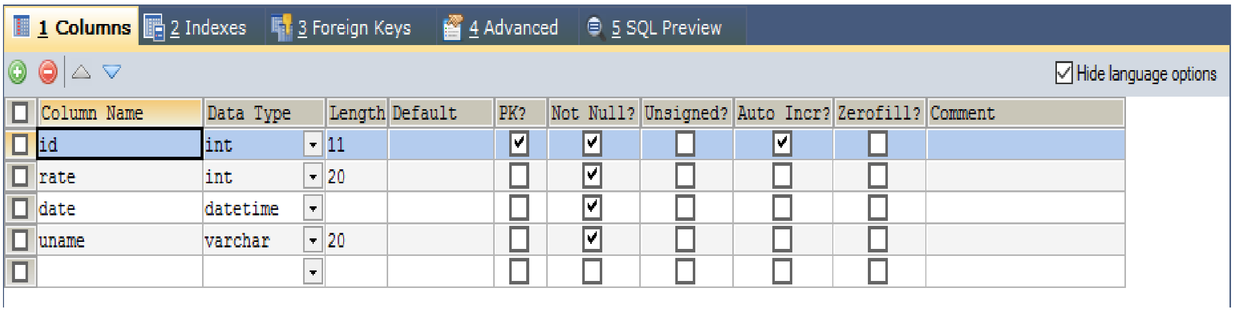
7.10 review_details



☑ Hide language options

□	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	Comment
□	id	int	11		☑	☑	□	☑	□	
□	review	varchar	500		□	☑	□	□	□	
□	date	datetime			□	☑	□	□	□	
□	uname	varchar	20		□	☑	□	□	□	
□					□	□	□	□	□	

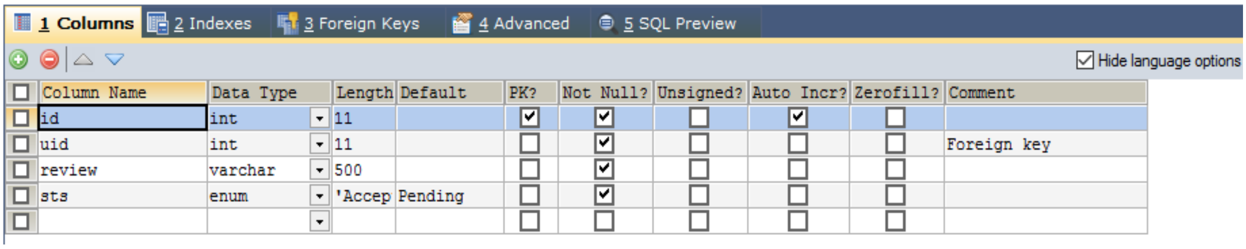
7.11 rating_details



☑ Hide language options

☐	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	Comment
☐	id	int	11		☑	☑	☐	☑	☐	
☐	rate	int	20		☐	☑	☐	☐	☐	
☐	date	datetime			☐	☑	☐	☐	☐	
☐	uname	varchar	20		☐	☑	☐	☐	☐	
☐					☐	☐	☐	☐	☐	

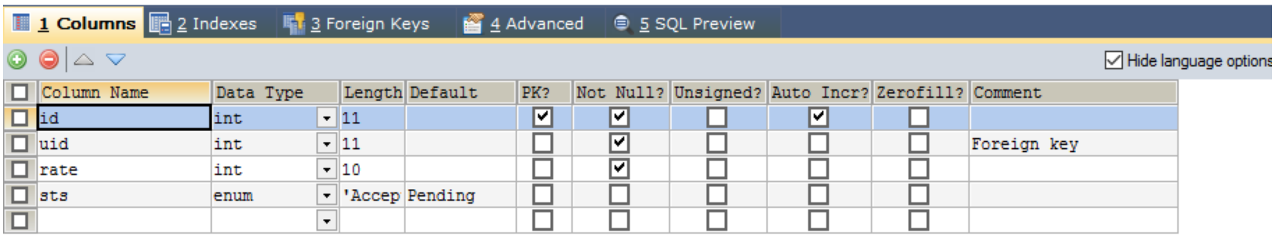
7.11 our_review



☑ Hide language options

☐	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	Comment
☐	id	int	11		☑	☑	☐	☑	☐	
☐	uid	int	11		☐	☑	☐	☐	☐	Foreign key
☐	review	varchar	500		☐	☑	☐	☐	☐	
☐	sts	enum	'Accep	Pending	☐	☑	☐	☐	☐	
☐					☐	☐	☐	☐	☐	

7.12 our_rating



☑ Hide language options

☐	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	Comment
☐	id	int	11		☑	☑	☐	☑	☐	
☐	uid	int	11		☐	☑	☐	☐	☐	Foreign key
☐	rate	int	10		☐	☑	☐	☐	☐	
☐	sts	enum	'Accep	Pending	☐	☐	☐	☐	☐	
☐					☐	☐	☐	☐	☐	

7. CONCLUSION

The proposed project shows the flow, structure and working of FairPlay. We have introduced FairPlay, a system to detect both fraudulent and malware Google Play apps. Our experiments on a newly contributed longitudinal app dataset, have shown that a high percentage of malware is involved in search rank fraud; both are accurately identified by FairPlay. In addition, we showed FairPlay's ability to discover hundreds of apps that evade Google Play's detection technology, including a new type of coercive fraud attack.Our system is efficient, reliable and time saving.

REFERENCES

[1]. Mahmudur Rahman, Mizanur Rahman, Bogdan Carbunar, Duen Horng Chau: Search Rank Fraud and Malware Detection in Google Play In *Proceedings of the IEE.IEEE 2017*

[2]. Yajin Zhou and Xuxian Jiang. Dissecting Android Malware: Characterization and Evolution. In *Proceedings of the IEEE S&P*,pages 95–109. IEEE, 2012.

[3]. Iker Burguera, Urko Zurutuza, and Simin Nadjm-Tehrani. Crowdroid: Behavior-Based Malware Detection System for Android. In *Proceedings of ACM SPSM*, pages 15–26. ACM, 2011.

[4]. Asaf Shabtai, Uri Kanonov, Yuval Elovici, Chanan Glezer, and Yael Weiss. Andromaly: a Behavioral Malware Detection Framework for Android Devices. *Intelligent Information Systems*, 38(1):161–190, 2012.

[5]. Ezra Siegel. Fake Reviews in Google Play and Apple App Store. Appentive, 2014.

[6]. Zach Miners. Report: Malware-infected Android apps spike in the Google Play store. PCWorld, 2014.

[7]. Leman Akoglu, Rishi Chandy, and Christos Faloutsos. Opinion Fraud Detection in Online Reviews by Network Effects. In *Proceedings of ICWSM*, 2013.