CHAPTER 1

INTRODUCTION

1. <u>INTRODUCTION</u>

In India exam is the most concerned issue for students and for teacher as well. To provide better education to students it is essential to evaluate them on the basis of their daily or weekly performance. As we have seen for the better development of student it is obligatory to provide them quality education in which new technologies as well new ideas have to be implemented.

In conventional Education System students are evaluated once in a 2 month. At that time it is difficult to examine the actual potential of students on the basis of these conformist exams. As we can see from these traditional exams, a student does not take these examinations seriously and get passed through these examinations without taking so much effort. So students are not benefited substantially by this education system.

In some advanced universities if the instructor wants to build the tendency of surprise test or daily evaluation test among the students, the challenge they confront is that it get more hectic to manage the short exam as well as lecture in a very short span of 60 minutes. In practical life it takes more than 20 minutes to conduct the short quiz on the paper and after that most time is consumed in evaluating the papers.

Moreover we feel like that in today's era of modern technological interventions, there is a need of such assets which evaluates and acknowledges the caliber of students in a very short span of time without any need of wasting paper and time resource. This can be very helpful for those corporate industries which want to evaluate the aptitude of a candidate for their selection.

- **Our project** – *"PAREEKSHA" (Android Application)* aims to solve the challenges faced by teachers during their class hours. This Application will assist the teachers to take short quiz randomly during classes.

Also, the Android is ruling the mobile app user base. This will help our respected teachers as well as our industries to build up the learning habit among the youth generation.

- Teacher (User) can conduct the exam by uploading their respecting subject's test paper.
- Student (User) can give the exam from anywhere at the same time.

The project aims to cover all aspects of education. The person who wants to conduct the exam using this app can do it in a simple and efficient way.

1.1 Motivation

The motivation behind developing this application is the very simple fact or the challenge that is faced by a teacher and student during the class room hours. Time is a very crucial resource that is not easily available to the teacher. A 40 minute lecture daily got wasted in taking attendance, and the time left is utilized in teaching, but in this whole process the academic acumen of a student got degraded since he is not regularly evaluated for his performance, and his understanding of the topic. More-over there is a need of an application which is portable to carry and a student can give exam anywhere he wants. This can be very helpful for organizing the competitive programs in colleges where this application can play the role of eliminating candidates by organizing a short quick test on their android phones.

1.2 Why Android Platform

Android platform is very friendly and comfortable as it is based on java programming which is easy to learn. Android platform has attracted many mobile users and Android programmers. Now the Android app platform is trust worthy as it is free from bugs and errors. Mobile apps are now catching up at a rapid pace whether it is a security application, fun applications, business applications, internet applications, utility applications and many more that can run on a smartphone platform. As users are multiplying immensely with the demands of both custom and customized applications for their respective devices, most of the leading smartphone manufacturers are playing major role to thrive in it. Android's SDK has been modified in such a way that it automatically finds bugs and tests the application. High performance and quality applications can be developed by using the SDK as it provides great built-in features as well as high level flexibility to develop smart mobile applications. So it is true that for innovative and far-reaching android apps development, we require highly skillful and experienced Android developers.

Android is a Linux-based operating system designed primarily for touchscreen mobile devices such as smartphones and tablet computers. Initially developed by Android Inc., which Google backed financially and later 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.

Android was our choice of development as in the current scenario, android is the popular choice of the user. Also cheap handsets from variety of manufacturers are available with inbuilt functions, sensors and quality hardware. 7 out of 10 android smartphones in India are below the price tag of 10000.When it comes to Android, the possibilities of choices are unlimited. There is a huge variety of devices to choose from across brands, sizes and price ranges. Android is a much more advanced and well thought out operating system compared to other platforms. It is open source and most of its applications are free.

1.3 Why PHP Framework

Rasmus Lerdorf wrote the first PHP - first called Personal Home Page - scripts as a series of Perl scripts that he used to track visitors to his webpage and to see who was viewing his resume. He eventually rewrote PHP as a scripting engine and added support for forms. Over the years the Personal Home Page acronym was dropped and it evolved into the PHP Hypertext. PHP can be used to create web applications ranging from personal websites to e-commerce applications and community web portals i.e. discussion forums, blogs etc. PHP is freely available for use. The community of open source PHP developers provides technical support and is constantly improving updating the core PHP functionalities. PHP is available at free of cost under PHP General Public License and most of its associative required software's like MySQL, Text Editors and Apache Server are also freely available, so it proves very cost effective for the developers.

PHP provides high compatibility with leading operating systems and web servers such as thereby enabling it to be easily deployed across several different platforms. PHP scripts can run across operating systems such as Linux, Windows, Solaris, OpenBSD, Mac OSX etc and also provide support for all major web servers such as Apache, IIS, iPlanet etc. Several web tasks can now be easily perform using PHP. For example now we can develop from small websites to giant business and organizational websites, informative forums, chatting platforms, CRM solutions, e-commerce shopping carts, community websites, e-business, shopping carts and gigantic database driven sites. Designed in a user friendly manner, PHP gives more flexibility than C, C++ and ASP and overall helps in increasing traffic to the site. PHP is designed to work well with the web, and so things like accessing the GET and POST and working with HTML and URLs are built-ins in the PHP language. This makes it really

concise and straightforward to make a website. Being an open source language, a large number of libraries and extensions, to extend its core functionalities, are available for download. The source code of PHP can be modified to include custom created extensions and components thereby increasing its extensibility. Nowadays developing dynamic websites are in the huge demand due to its specific characteristics like it automatically refreshes and does not need to make many changes manually. A huge advantage that PHP offers is its community. If you are looking for a particular script, chances are another user has already created something similar. Check within the PHP community for availability. Likewise, if you have created a function that others might enjoy, be sure to post the code for others. If you need to access other web based tools like Google maps (which is always advisable for a business website), or any other, PHP makes it easy to access. PHP offers security as well that helps prevent malicious attacks. These security levels can be adjusted in the .ini file.

CHAPTER 2

REVIEW OF LITERATURE

2. REVIEW OF LITERATURE

2.1 Proposed application(s)

The proposed system consists of three different software bundles described as below.

1.   **Client Android App:** The android app is the software that runs on the Android OS and the client side. It must have following functionalities:
   - Provide login support for teachers and students.
   - Teacher can upload the test papers with unique TESTid & code and update the database accordingly.
   - Implement new element like attendance dashboard.
   - Student can give any exam with predefined knowledge of TESTid & code.
   - Student can see the marks obtained at the end of examination.
   - Exam can be organized on local Wi-Fi server without any use of internet.

2.   **Web Connectivity:** The web connectivity secures a connection between the application and the web server which can be accessed remotely. It must include these features:
   - Provide links for teacher to upload the test paper easily.
   - Provide an interface for anyone who wants to conduct the exam.

3.   **The Backend server**: The backend server is required to cater the queries generated by the android application. Following features are desirable:
   - Server must exist on cloud for the internet services. It must act as a backend to the web site.
   - To conduct the exam, a connection on local Wi-Fi server must exist on the local machine.
   - It should cater the requests of the android application.

CHAPTER 3

PROBLEM SPECIFICATION

3. Problem Specification

The main objective of building this application is to:

- Organize a short exam instantly anywhere, anytime.
- The application must occupy very less memory on the smart-phone for its fast execution.
- The application must provide a simple interface for interaction of the application with both teacher and student.
- Teacher must be provided with the admin privileges for conducting exam.
- Teacher must be able to see, edit and delete the question paper.
- A student can review his marks.
- The application must run in a minimum internet speed.
- Most importantly the application should run offline also with the help of Wi-Fi connectivity.

CHAPTER 4

REQUIREMENT ANALYSIS

4 <u>REQUIREMENT ANALYSIS AND FEASIBILITY</u>

4.1 Software Requirements

4.1.1. Client side android application

Operating System	:	Android v2.3.3
Pre-installed applications	:	Google Play Services apk installed

4.1.2. Web Connectivity

Operating system	:	Any one that supports internet connection
Browser	:	IE 9.0+, Firefox 10.0+, Chrome 15.0+
Graphic acceleration	:	Preferable

4.1.3. Server end application

Operating system	:	Linux
MySql database	:	MySql 5.5+
Xampp server	:	Xampp 1.8.1+

4.2 Hardware Requirements

4.2.1 Client side android application

Processor	:	600 MHz and above
Free Space	:	5 MB
RAM	:	256MB
Internet Connectivity	:	At least 2G

4.2.2. Web Connectivity

RAM	:	minimum 256 MB
Graphic card	:	Required
Internet connectivity	:	Required
Processor	:	600 MHz and above

4.2.3. Server end application

RAM (one cloudlet)	:	128MB
Processor (one cloudlet)	:	200 MHz
Internet connectivity	:	High speed, high bandwidth internet connection

4.3 Feasibility Study

4.3.1. Technical feasibility

Evaluating the technical feasibility in our application *PAREEKSHA* is the trickiest part of a feasibility study as with the growing need of the user and growing innovation in the field of mobile computing and cloud computing. This is because, at this point of time when the user is in alpha testing mode, not too many detailed design of the system exists, making it difficult to access issues like performance, costs on (on account of the kind of technology to be deployed) instructor. A number of issues have to be considered while doing a technical analysis.

**Understanding the different technologies involved in the proposed system*:*

Before commencing the project, we have to be very clear about what are the technologies that are to be required for the development of the new system.

There always exists a confusion between user's need and programmer's choice. The best option between the two needs has to be carried out.

**Find out whether the organization currently possesses the required technologies*:*

- Is the required technology available with the organization?

- Will the current technology in the hands of the ends user sufficiently meets all the requirements of the application or do we need to add more hardware / software services to provide the optimal output?
- If so is the capacity sufficient?
- Will the current cloud database system able to handle the queries generated by the system and able to scale properly and easily reply to the response cycle from the end user?

4.3.2 Operational Feasibility

Proposed project is beneficial only if it can be turned into educational assets that will meet the organizations operating requirements. Simply stated, this test of feasibility asks if the system will work when it is developed and installed. Are there major barriers to Implementation? Here are questions that will help test the operational feasibility of a project: Is there sufficient support for the project from management from users? If the current system is well liked and used to the extent that persons will not be able to see reasons for change, there may be resistance.

Are the current business methods acceptable to the user? If they are not, Users may welcome a change that will bring about a more operational and useful systems. Have the user been involved in the planning and development of the project?

Early involvement reduces the chances of resistance to the system and in general and increases the likelihood of successful project.

Since the inception of this project it has been able to operate feasibly and work smoothly, there were several hardships in the automation of several modules but once they were dealt with then the project worked off well.

4.3.3. Economic Feasibility:

Economic feasibility attempts to weigh the costs of developing and implementing a new system, against the benefits that would accrue from having the new system in place. This feasibility study gives the top management the economic justification for the new system.
A simple economic analysis which gives the actual comparison of costs and benefits are much more meaningful in this case. In addition, this proves to be a useful point of reference to compare actual costs as the project progresses. There could be various types of intangible benefits on account of automation. These could include increased customer satisfaction, improvement in product quality better decision making timeliness of information, expediting activities, improved accuracy of operations, better documentation and record keeping, faster retrieval of information, better employee morale.

For developing *PAREEKSHA* as the whole project is based on free and open source software such as Android SDK, PHP there was not much cost in the development, but for the deployment of the app in the cloud cost factor comes since with the increasing cloudlets the cost increases so economically only server costs adds up to it.

CHAPTER 5

TOOLS
&
TECHNOLOGY

5. TOOLS AND TECHNOLOGY USED

The list of all the tools and technologies used by us in the development of the project are:

- Google Android OS 2.3.3 for application platform
- PHP Framework
- My SQL 5.5
- HTML
- CSS
- XAMPP Server
- Google play services
- Virtual Router

5.1 Introduction to Google Android

5.1.1 What is Google Android?

Android is an operating system based on Linux with a Java programming interface. It allows background processing, provides a rich user interface library, supports 2D and 3D graphics using the OpenGL libraries, access to the file system and provides an embedded SQLite database.

Google, famous for its dominating web search portal, entered the market with a Linux kernel-based product: Google Android. Google Android, as the new contestant on the market, gained great popularity after being first announced in November 2007. In order to guarantee a great impact of Google Android on the market, Google organized several developer contests in order to create a rich set of ready-to-go applications for the Android platform before the actual phone is even available on the market.

Basically, Android consists of a UNIX-like operating system based on a 2.6- series Linux kernel. The operating system has been enriched with all necessary elements that are required to provide basic functions like a networking stack, GSM/GPRS abilities, and more. On top

of all this, Google offers a framework containing a rich set of Java methods enabling developers to create a wide range of software for mobile use on Google Android.

5.1.2 Google Android Architecture
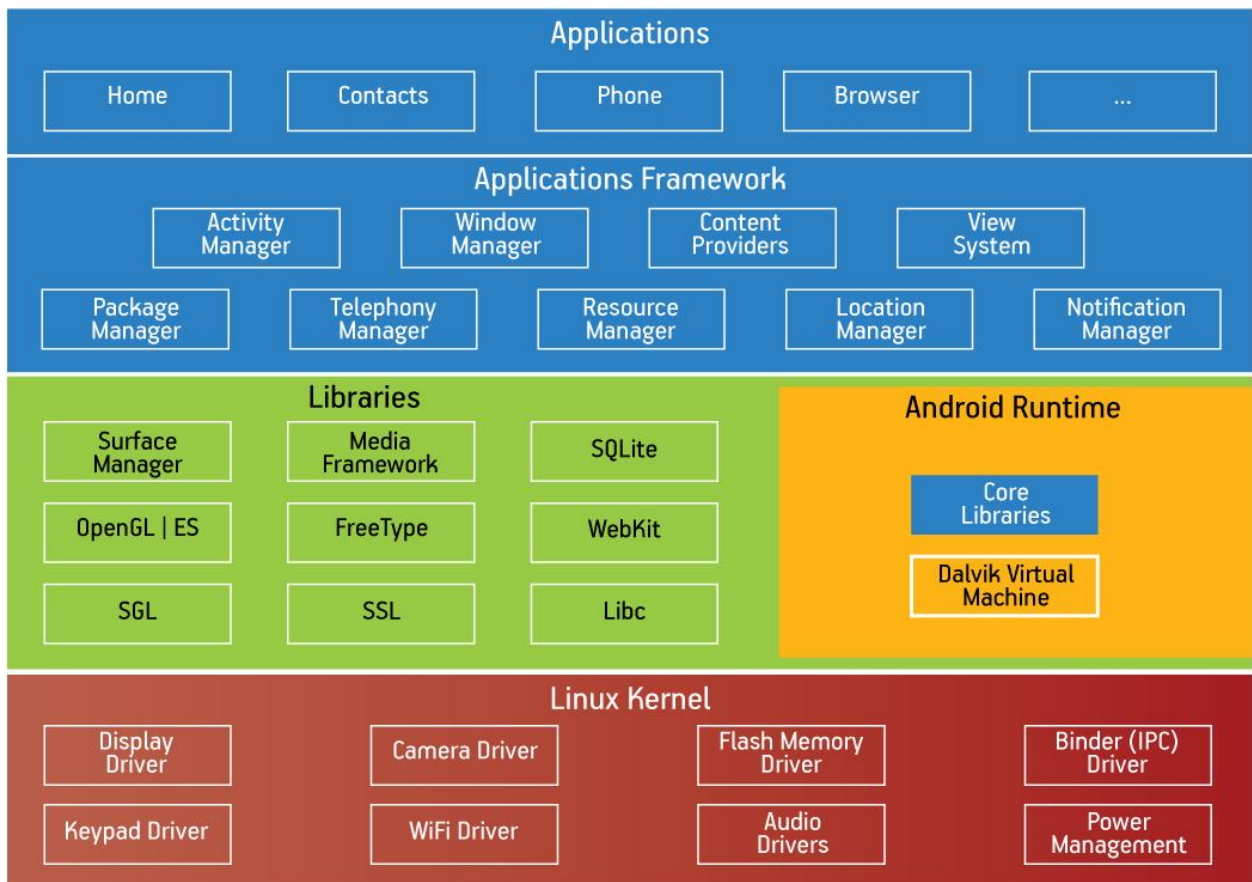
Fig. 5.1- Android Architecture

Google usually refers to the Android OS as a software stack. Each layer of the stack groups together several programs that support specific system functions.

The base of the stack is the kernel. Google used the Linux version 2.6 OS to build Android's kernel, which includes Android's memory management programs, security settings, power management software and several hardware drivers. Drivers are programs that control hardware

devices. For example, the Nexus One has a camera. The Android kernel includes a camera driver, which allows the user to send commands to the camera hardware.

The next level of software includes Android's libraries. You can think of libraries as a set of instructions that tell the device how to handle different kinds of data. For example, the media framework library supports playback and recording of various audio, video and picture formats. Other libraries include a three-dimensional acceleration library (for devices with accelerometers) and a Web browser library.

Located on the same level as the libraries layer, the Android runtime layer includes a set of core Java libraries -- Android application programmers build their apps using the Java programming language.

A virtual machine is a software application that behaves as if it were an independent device with its own operating system. You can run a virtual machine on a computer that operates on a completely different OS than the physical machine's OS. The Android OS uses virtual machines to run each application as its own process. That's important for a few reasons. First, no application is dependent upon another. Second, if an application crashes, it shouldn't affect any other applications running on the device. Third, it simplifies memory management.

The next layer is the application framework. This includes the programs that manage the phone's basic functions like resource allocation, telephone applications, switching between processes or programs and keeping track of the phone's physical location. Application developers have full access to Android's application framework. This allows them to take advantage of Android's processing capabilities and support features when building an Android application. Think of the application framework as a set of basic tools with which a developer can build much more complex tools.

At the top of the stack are the applications themselves. This is where you find the basic functions of the device such as making phone calls, accessing the Web browser and accessing your contacts list. If you're an average user, this is the layer you'll use most. You do that with the user interface. Only Google programmers, application developers and hardware manufacturers access the other layers further down the stack.

5.1.3 The Google Android SDK

As no officially obtainable physical device is available at time of writing, we will have to rely on the provided development kit: Google Android SDK.

A core element of the SDK is the actual Google Android Emulator which provides a graphical emulation of a possible handheld device running Google Android. Furthermore, the SDK not only provides the core classes of the Android framework packed into a Java Jar-file: it also includes the documentation in HTML-form and several tools that improve the usability and interaction with the emulator.

To make things easier, Google suggests Eclipse as the IDE to be used. For this, Google optionally provides an Android Plug-in made for Eclipse that will take over a few basic tasks like creating the Android Project layout on disk, the integration of your SDK, eventually starting your emulator if not running yet and the deployment of you application on the emulator. The Android SDK provides you the API libraries and developer tools necessary to build, test, and debug apps for Android.

5.1.4 Android Versions

The first commercial version, Android 1.0, was released in September 2008 then second version was Android 1.1 released in development by Google and the Open Handset Alliance (OHA), and has seen a number of updates to its base operating system since its original release. These updates typically fix bugs and add new features.

Since April 2009, Android versions have been developed under a codename and released in alphabetical order:

- **Cupcake (Android 1.5)**

Technically Android 1.5 wasn't the first version, but versions before that doesn't seem to have received any codenames. Stories were told that it was supposed to be version 1.2, but Google decided to make it a major revision and made it 1.5 instead, and gave it the dessert name "cupcake" as a codename, and that's when the "dessert series" got started.

A cupcake is a small cake, the size of an individual portion, baked in a cup-shaped mold, usually served with frosting on top.

- **Donut (Android 1.6)**

Android V1.6, codename Donut, was released in September 2009. It fixed reboot errors in the OS as well as revamped photo and video features (i.e. camera interface) and better search integration. It also added support for larger screen size, and is the first version to offer Google turn-by-turn navigation feature.

Donut is usually defined as a small ring-shaped fried cake, is actually spelled doughnut, made out of rich, light dough and deep fried, then various sweet coating can be added

- **Eclair (Android 2.0, 2.0.1, 2.1)**

Android 2.0 was released in October 2009, with a bugfix version 2.0.1 in December 2009. Android 2.1 was released January of 2010. Most people consider them a single release. Added features include Bluetooth 2.1 support; flash and digital zoom for the camera, multi-touch support, live wallpapers, and more.

Eclair the dessert is usually describe as an oblong "cream puff", a baked pastry with cream filling and chocolate coating on top.

- **Froyo (Android 2.2 - 2.2.3)**

Android 2.2 mainly improved speed by adopting the Javascript "just-in-time" compiler engine from the Google browser "Chrome", as well as improved browser support by adding animated GIF support, Flash 10.1 plug-in support, along with USB tethering and Wi-Fi Hotspot capability (for those with supporting hardware).

Froyo is actually concatenated from "frozen yogurt", but is also a brand name. It is a frozen dessert made from yogurt, so it is slightly sourer, but also lowers in fat.

- **Gingerbread (2.3-2.3.2 & 2.3.3- 2.3.7)**

Gingerbread man on Google campus has been reported in early November 2010, and Gingerbread is officially released in December 2010.

On December 7th, 2010, Google officially announced the Nexus S (made by Samsung) to be released on T-Mobile, the first phone with Android OS 2.3 Gingerbread. SDK has been released. Nexus S has since also appeared on Sprint and AT&T.

Gingerbread features support for SIP internet calling, NFC wireless transaction capability (if hardware is present), support for more than one camera, support for gyroscope and other sensors (barometer, gravimeter, and others are possible), download manager, some tweaks to allow usage on Tablets, and other system level tweaks for programmers.

In desserts, gingerbread is basically ginger-flavored "cake" or "cookie". It is often made to celebrate end-of-year holidays in the US, and often in the shape of a man.

- **Honeycomb (Android 3.0,3.1,3.2)**

Honeycomb was released in February 2011, and was rapidly followed by 3.1 and 3.2 which only came out in July / August 2011. Android-based tablets should be running 3.X as that will be optimized for tablets, (implying current Android OS 2.X are not really designed for tablets) That did not stop a slew of smaller manufacturers as well as Samsung from putting out an army of Android tablets of various sizes before the end of 2010 with Android 2.X hoping to catch the Christmas shopping season and success of Apple's iPad.

Motorola Xoom is the first Android 3.X tablet to be released, It has since been followed by a ton of Tablets.

Dessert-wise Honeycomb is a sweetened breakfast cereal that has been made since 1965 by Post Cereals. It consists of honey-flavored corn cereal bits in a honeycomb shape.

- **Ice Cream Sandwich (Android 4.0-4.2 & 4.0.3-4.0.4)**

It appears now that Honeycomb (3.X) will remain a tablet-only OS, while the phones will keep getting Gingerbread (2.x) updates until the unified codebase, i.e. Android OS 4.0, which will be "Ice Cream Sandwich" launch some time in the future. In real life, ice cream sandwich is a layer of ice cream, usually vanilla, sandwiched between two chocolate cookies, and usually rectangular in shape.

- **Jelly Bean (Android 4.1, 4.3)**

On June 26th, the Jelly Bean mascot appeared on Google campus.

- **KitKat (Android 4.4,4.4.3)**

Google has JUST announced on 03-SEP-2013 that Android 4.4 will be named Kitkat. Apparently Nestle is fully onboard, as they have launched an advertising campaign with Google. Specially marked packages of Kitkat will contain a sweepstakes code that can win you a new Nexus 7 Android tablet, or Google Play store credit.

5.2 PHP Framework

5.2.1 Introduction to PHP

PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. As of January 2013, PHP was installed on more than 240 million websites (39% of those sampled) and 2.1 million web servers. Originally created by Rasmus Lerdorf in 1994, the reference implementation of PHP is now produced by The PHP Group. While PHP originally stood for Personal Home Page, it now stands for PHP: Hypertext Preprocessor, a recursive acronym.

PHP code can be simply mixed with HTML code, or it can be used in combination with various templating engines and web frameworks. PHP code is usually processed by a PHP interpreter, which is usually implemented as a web server's native module or common (CGI) executable. After the PHP code is interpreted and executed, the web server sends resulting output to its client, usually in form of a part of the generated web page – for example, PHP code can generate a web page's HTML code, an image, or some other data. PHP has also evolved to include a command-line interface (CLI) capability and can be used in standalone graphical.

PHP is free software released under the PHP License. PHP has been widely ported and can be deployed on most web servers on almost every operating system and platform, free of charge

PHP development began in 1994 when the developer Rasmus Lerdorf wrote a series of Common Gateway Interface (CGI)Perl scripts, which he used to maintain his personal homepage. The tools performed tasks such as displaying his résumé and recording his web traffic. He rewrote these scripts in C for performance reasons, extending them to add the ability to work with web forms and to communicate with databases, and called this implementation "Personal Home Page/Forms Interpreter" or PHP/FI.

PHP/FI could be used to build simple, dynamic web applications. Lerdorf initially announced the release of PHP/FI as "Personal Home Page Tools (PHP Tools) version 1.0" publicly to accelerate bug location and improve the code, on the Usenet discussion group on June 8, 1995. This release already had the basic functionality that PHP has as of 2013. This included

Perl-like variables, form handling, and the ability to embed HTML. The syntax resembled that of Perl but was simpler, more limited and less consistent.

Early PHP was not intended to be a new programming language, and grew organically, with Lerdorf noting in retrospect: "I don't know how to stop it, there was never any intent to write a programming language. I have absolutely no idea how to write a programming language, I just kept adding the next logical step on the way." A development team began to form and, after months of work and beta testing, officially released PHP/FI 2 in November 1997.

One criticism of PHP is that it was not originally designed, but instead it was developed organically; among other things, this has led to inconsistent naming of functions and inconsistent ordering of their parameters. In some cases, the function names were chosen to match the lower-level libraries which PHP was "wrapping", while in some very early versions of PHP the length of the function names was used internally as a hash function, so names were chosen to improve the distribution of hash values.

Zeev Suraski and Andi Gutmans rewrote the parser in 1997 and formed the base of PHP 3, changing the language's name to the recursive acronym PHP: Hypertext Preprocessor. Afterwards, public testing of PHP 3 began, and the official launch came in June 1998. Suraski and Gutmans then started a new rewrite of PHP's core, producing the Zend Engine in 1999. They also founded Zend Technologies in Ramat Gan, Israel.

On May 22, 2000, PHP 4, powered by the Zend Engine 1.0, was released. As of August 2008 this branch reached version 4.4.9. PHP 4 is no longer under development nor will any security updates be released.

On July 13, 2004, PHP 5 was released, powered by the new Zend Engine II. PHP 5 included new features such as improved support for object-oriented programming, the PHP Data Objects (PDO) extension (which defines a lightweight and consistent interface for accessing databases), and numerous performance enhancements. In 2008 PHP 5 became the only stable version under development. Late static binding had been missing from PHP and was added in version 5.3.

Many high-profile open-source projects ceased to support PHP 4 in new code as of February 5, 2008, because of the GoPHP5 initiative, provided by a consortium of PHP developers promoting the transition from PHP 4 to PHP 5.

PHP interpreters are available on most existing 32-bit and 64-bit operating systems, either by building them from the PHP source code, or by using pre-built binaries. For the PHP versions 5.3 and 5.4, the only available Microsoft Windows binary distributions were 32-bit x86 builds, requiring Windows 32-bit compatibility mode while using Internet Information Services (IIS) on a 64-bit Windows platform. PHP version 5.5 made the 64-bit x86-64 builds available for Microsoft Windows.

5.3 Overview of MySQL

5.3.1. Introduction

MySQL, the most popular Open Source SQL database management system, is developed, distributed, and supported by Oracle Corporation. It started out with the intention of using the MySQL database system to connect to our tables using our own fast low-level (ISAM) routines. However, after some testing, we came to the conclusion that MySQL was not fast enough or flexible enough for our needs. This resulted in a new SQL interface to our database but with almost the same API interface as MySQL. This API was designed to allow third-party code that was written for use with MySQL to be ported easily for use with MySQL.

The name of the MySQL Dolphin is ―Sakila, which was chosen from a huge list of names suggested by users in our ―Name the Dolphin contest. The winning name was submitted by Ambrose Twebaze, an Open Source software developer from Swaziland, Africa. According to Ambrose, the feminine name Sakila has its roots in SiSwati, the local language of Swaziland. Sakila is also the name of a town in Arusha, Tanzania, near Ambrose's country of origin, Uganda.

- **MySQL is a database management system.**

A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or the vast amounts of information in a corporate network. To add, access, and process data stored in a computer database, you need a database management

system such as MySQL Server. Since computers are very good at handling large amounts of data, database management systems play a central role in computing, as standalone utilities, or as parts of other applications.

- **MySQL databases are relational.**

A relational database stores data in separate tables rather than putting all the data in one big storeroom. The database structures are organized into physical files optimized for speed. The logical model, with objects such as databases, tables, views, rows, and columns, offers a flexible programming environment. You set up rules governing the relationships between different data fields, such as one-to-one, one-to-many, unique, required or optional, and ―pointers‖ between different tables. The database enforces these rules, so that with a well-designed database, your application never sees inconsistent, duplicate, orphan, out-of-date, or missing data.

The SQL part of ―MySQL stands for ―Structured Query Language. SQL is the most common standardized language used to access databases. Depending on your programming environment, you might enter SQL directly (for example, to generate reports), embed SQL statements into code written in another language, or use a language-specific API that hides the SQL syntax.

SQL is defined by the ANSI/ISO SQL Standard. The SQL standard has been evolving since 1986 and several versions exist. In this manual, ―SQL-92 refers to the standard released in 1992, ―SQL:1999 refers to the standard released in 1999, and ―SQL:2003 refers to the current version of the standard. We use the phrase ―the SQL standard to mean the current version of the SQL Standard at any time.

- **MySQL software is Open Source**

Open Source means that it is possible for anyone to use and modify the software. Anybody can download the MySQL software from the Internet and use it without paying anything. If you wish, you may study the source code and change it to suit your needs. The MySQL software uses the GPL (GNU General Public License), http://www.fsf.org/licenses/ , to define what you may and may not do with the software in different situations. If you feel

uncomfortable with the GPL or need to embed MySQL code into a commercial application, you can buy a commercially licensed version from us.

- **The MySQL Database Server is very fast, reliable, scalable, and easy to use.**

If that is what you are looking for, you should give it a try. MySQL Server can run comfortably on a desktop or laptop, alongside your other applications, web servers, and so on, requiring little or no attention. If you dedicate an entire machine to MySQL, you can adjust the settings to take advantage of all the memory, CPU power, and I/O capacity available. MySQL can also scale up to clusters of machines, networked together. MySQL Server was originally developed to handle large databases much faster than existing solutions and has been successfully used in highly demanding production environments for several years. Although under constant development, MySQL Server today offers a rich and useful set of functions. Its connectivity, speed, and security make MySQL Server highly suited for accessing databases on the Internet.

- **MySQL Server works in client/server or embedded systems.**

The MySQL Database Software is a client/server system that consists of a multithreaded SQL server that supports different backends, several different client programs and libraries, administrative tools, and a wide range of application programming interfaces (APIs).

We also provide MySQL Server as an embedded multi-threaded library that you can link into your application to get a smaller, faster, easier-to-manage standalone product.

- **A large amount of contributed MySQL software is available.**

MSQL Server has a practical set of features developed in close cooperation with our users. It is very likely that your favorite application or language supports the MySQL Database Server.

5.3.2 Main Features of MySQL

5.3.2.1 Internals and Portability:

- Written in C and C++.

- Tested with a broad range of different compilers.
- Works on many different platforms.
- For portability, uses CMake in MySQL 5.5 and up. Previous series use GNU Automake, Autoconf, and Libtool.
- Tested with Purify (a commercial memory leakage detector) as well as with Valgrind, a GPL tool.
- Uses multi-layered server design with independent modules.

- Designed to be fully multi-threaded using kernel threads, to easily use multiple CPUs if they are available.

- Provides transactional and non-transactional storage engines. Uses very fast B-tree disk tables (MyISAM) with index compression.

- Designed to make it relatively easy to add other storage engines. This is useful if you want to provide an SQL interface for an in-house database.

- Uses a very fast thread-based memory allocation system.

- Implements in-memory hash tables, which are used as temporary tables.

- Implements SQL functions using a highly optimized class library that should be as fast as possible. Usually there is no memory allocation at all after query initialization.

- Provides the server as a separate program for use in a client/server networked environment, and as a library that can be embedded (linked) into standalone applications. Such applications can be used in isolation or in environments where no network is available.

5.3.2.2 Data Types:

- Many data types: signed/unsigned integers 1, 2, 3, 4, and 8 bytes long, FLOAT, DOUBLE, CHAR, VARCHAR, BINARY,VARBINARY, TEXT, BLO B, DATE, TIME, DATETIME, TIMESTAMP, YEAR, SET, ENUM, and OpenGIS spatial types.
- Fixed-length and variable-length string types.

5.3.2.3 Statements and Functions:

- Full operator and function support in the SELECT list and WHERE clause of queries.

- Full support for SQL GROUP BY and ORDER BY clauses. Support for group functions(COUNT(), AVG(), STD(),SUM(), MAX(), MIN(),and GROUP_CONCAT()).
- Support for LEFT OUTER JOIN and RIGHT OUTER JOIN with both standard SQL and ODBC syntax.
- Support for aliases on tables and columns as required by standard SQL.
- Support for DELETE, INSERT, REPLACE and UPDATE to return the number of rows that were changed (affected), or to return the number of rows matched instead by setting a flag when connecting to the server.
- Support for MySQL-specific SHOW statements that retrieve information about databases, storage engines, tables, and indexes. MySQL 5.0 adds support for the INFORMATION_SCHEMA database, implemented according to standard SQL.
- Independence of function names from table or column names. For example, ABS is a valid column name.

5.3.2.4 Security:

- A privilege and password system that is very flexible and secure, and that enables host-based verification.

- Password security by encryption of all password traffic when you connect to a server.

5.4 Overview of HTML

5.4.1 The Development of HTML

Berners-Lee developed and defined the HTML language, which was created and defined using SGML, during the development cycle for the first Web browser/editor from October to December 1990. The first version of the browser initially ran only on the NeXT platform and was only processing text files, but it was a start. Berners-Lee later put the code and specifications for the project (including HTML) on the Internet in the summer of 1991. During the next few years the system introduced by Berners-Lee caught on in the Internet community - and the 'web' of documents available was steadily growing. A common library of code was available to programmers to easily create the needed capabilities to access web documents. Browsers quickly became available for a wide variety of platforms. As the number of implementations grew, the variety did also. The HTML language originally specified by Berners-Lee had developed and extended far beyond its initial form and no real standard had yet been developed.

HTML, initials of Hypertext Markup Language, is the predominant markup language for web pages. It provides a means to describe the structure of text-based information in a document — by denoting certain text as headings, paragraphs, lists, and so on — and to supplement that text with interactive forms, embedded images, and other objects. HTML is written in the form of labels (known as tags), surrounded by angle brackets. HTML can also describe, to some degree, the appearance and semantics of a document, and can include embedded scripting language code which can affect the behavior of web browsers and other HTML processors.

HTML is also often used to refer to content of the MIME type text/html or even more broadly as a generic term for HTML whether in its XML-descended form (such as XHTML 1.0 and later) or its form descended directly from SGML.

HTML can be used to display any type of document on the host computer, which can be geographically at a different location. It is a versatile language and can be used on any platform or desktop.

5.5 Google Play Services (Android Market)

5.5.1 Introduction

Google Play Services give your apps more features to attract users on a wider range of devices. With Google Play services, your app can take advantage of the latest, Google powered features such as Maps, Google+, and more, with automatic platform updates distributed as an APK through the Google Play store. This makes it faster for your users to receive updates and easier for you to integrate the newest that Google has to offer.

Google Play services provide you with easy access to Google services and are tightly integrated with the Android OS. Easy-to-use client libraries are provided for each service that let you implement the functionality you want easier and faster.

All products in Google Play services share a common authorization API that leverages the existing Google accounts on the device. You and your users have a consistent and safe way to grant and receive OAuth2 access tokens to Google services.

Devices running Android 2.2 and newer and that have the Google Play Store app automatically receive updates to Google Play services. Enhance your app with the most recent version of Google Play services without worrying about your users' Android version.

The client library contains the interfaces to the individual Google services and allows you to obtain authorization from users to gain access to these services with their credentials. It also contains APIs that allow you to resolve any issues at runtime, such as a missing, disabled, or out-of-date Google Play services APK. The client library has a light footprint if you use ProGuard as part of your build process, so it won't have an adverse impact on your app's file size. If you want to access added features or products, you can upgrade to a new version of the

client library as they are released. However, upgrading is not necessary if you don't care about new features or bug fixes. We anticipate more Google services to be continuously added, so be on the lookout for these updates.

The Google Play services APK contains the individual Google services and runs as a background service in the Android OS. You interact with the background service through the client library and the service carries out the actions on your behalf. An easy-to-use authorization flow is also provided to gain access to the each Google service, which provides consistency for both you and your users.

The Google Play services APK is delivered through the Google Play Store, so updates to the services are not dependent on carrier or OEM system image updates. In general, devices running Android 2.3 (Gingerbread) or later and have the Google Play Store app installed receive updates within a few days. This allows you to use the newest APIs in Google Play services and reach most of the devices in the Android ecosystem.

Fig. 5.2: Google Play Services (APK)

CHAPTER 6

DESIGN

6. Design

The system after careful analysis has been identified to be presented with the following modules:

- Client Android Application
- Web Connectivity
- Backend Server
- Wi-Fi Connectivity

6.1 Client Android Application

Client Android Application developed in java programming language provides all the targeted features for users.

6.2 Web Connectivity

The web connectivity secures a presence of the application on internet which can be accessed widely. It must include these features:

- Provide links for teacher to upload the test paper easily.
- Provide an interface for anyone who wants to conduct the exam in quick manner.

6.3 Backend server

Backend server developed using PHP provides access to database on cloud for android to the client. The backend server has no GUI. Android application uses the request/reply architecture to interact with the database.

6.4 Data Flow Diagram

A graphical tool used to describe and analyze the moment of data through a system manual or automated including the process, stores of data, and delays in the system. Data Flow Diagrams are the central tool and the basis from which other components are developed. The transformation of data from input to output, through processes, may be described logically and independently of the physical components associated with the system. The DFD is also known as a data flow graph or a bubble chart.

DFDs are the model of the proposed system. They clearly should show the requirements on which the new system should be built. Later during design activity this is taken as the basis for drawing the system's structure charts. The Basic Notation used to create a DFD's are as follows:

- **Dataflow:** Data move in a specific direction from an origin to a   destination.
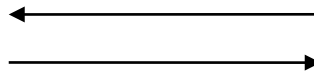
- **Process:** People, procedures, or devices that use or produce (Transform) Data. The physical component is not identified.
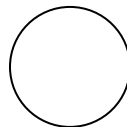
- **Source:** External sources or destination of data, which may be People, programs, organizations or other entities.
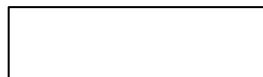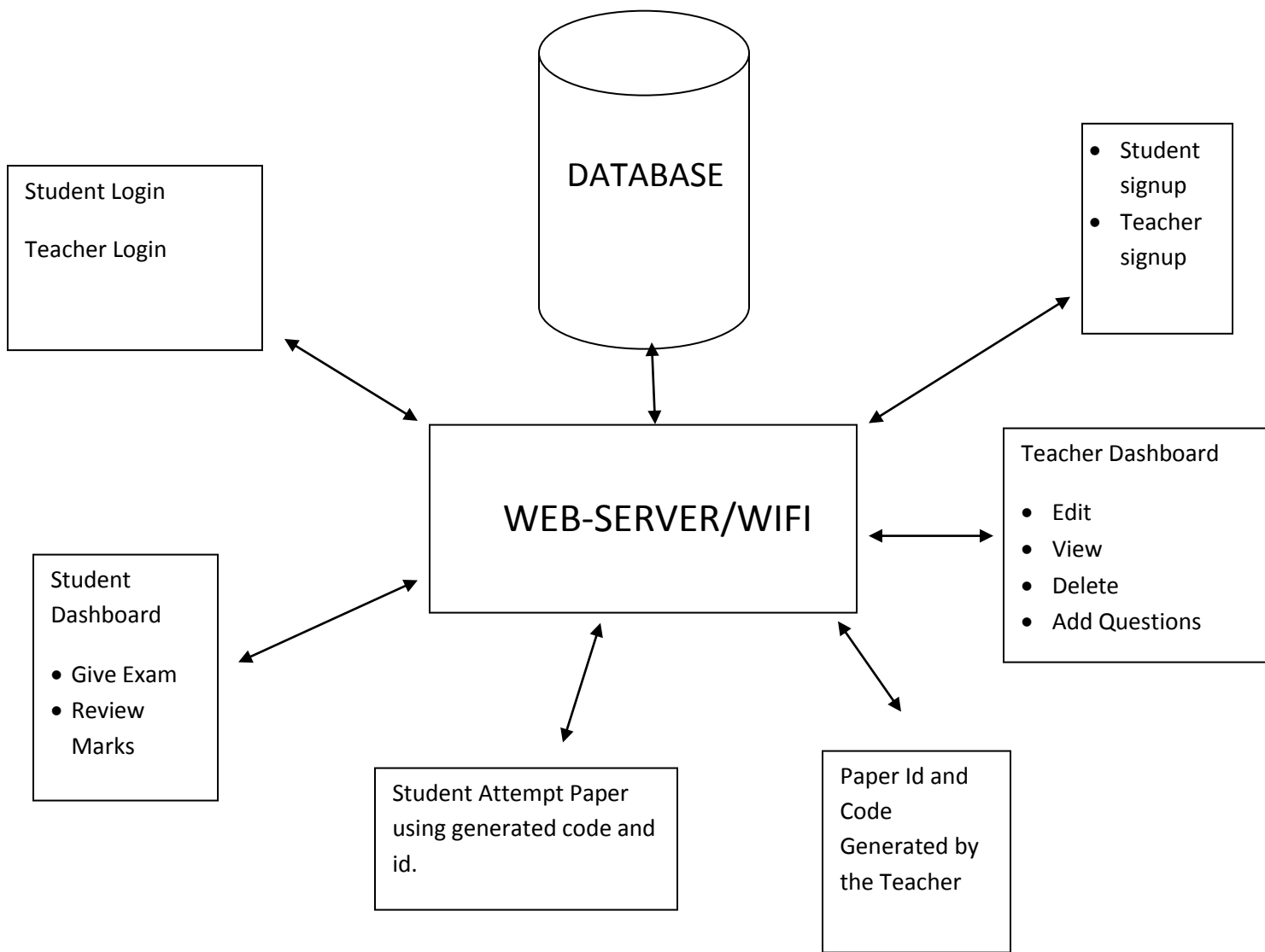
Fig. 6.1: Data Flow Diagram

CHAPTER 7

IMPLEMENTATION

7. <u>IMPLEMENTATION</u>

Fig. 7.1: Implementation Details

The implementation level detail of each module is given below:

7.1 Client Android Application:

The android application runs on android runtime Gingerbread 2.3. For using this app the user technically needs only an active internet connection on their mobile devices or local Wi-Fi access point. For conducting examination offline, we require a dedicated local server with enabled Wi-Fi access points. After the installation of the application on the smart-phones user find two components of our application.

- **Student**
- **Teacher**

7.1.1 Teacher Login

After the teacher sign up into the *PAREEKSHA* application, he/she has many possible options visible to   him/her:

- **Add Question***:* In this component, the teacher can see an add question button, through which he/she can add questions to the paper or can create a new paper. After the successful completion of question paper a unique paper ID will be generated through which student can identify their respective test. The question added by the user is automatically added to the database using android database connectivity.

- **Edit:** If a teacher wants to edit the question or he want to change the options of question of an already built paper, then he can edit it by this activity**.** The questions edited by the teacher were editing in the database automatically using android database connectivity.

- **Delete:** This option or button is available to the teacher if he wants to delete any question or the whole question paper in case he finds out any question as a doubtful one or wrong one.

- **View:** Teacher can view all the students who have given the exams along with their id and marks.

7.1.2 Student Login

After the Student sign up into the *PAREEKSHA* application, he/she has many possible options visible to   him/her:

- **Take Exam***:* Student can give the test of their subject with predefined knowledge of Test ID & code. Examination platform that is visible to the student is a simple platform with each page containing a single multiple choice question having four or many possible options. Using next button a student can navigate through various questions at a time. A student can

leave the exam in the middle of the exam at any time using this option of leaving (leave button).

- **View Marks:** Student can view the marks at end of examination or he can view the marks at any time with unique Test ID & Code. The marks are evaluated strictly on the basis of the marks set by the teacher at the time of creation of question paper. Negative marks can also be given to the student as a penalty.

7.2 Backend Server

Backend server developed using PHP provides access to database on cloud for android to the client. The backend server has no GUI. Android application uses the request/reply architecture to interact with the database.

7.3 Application Screenshots (Module wise)
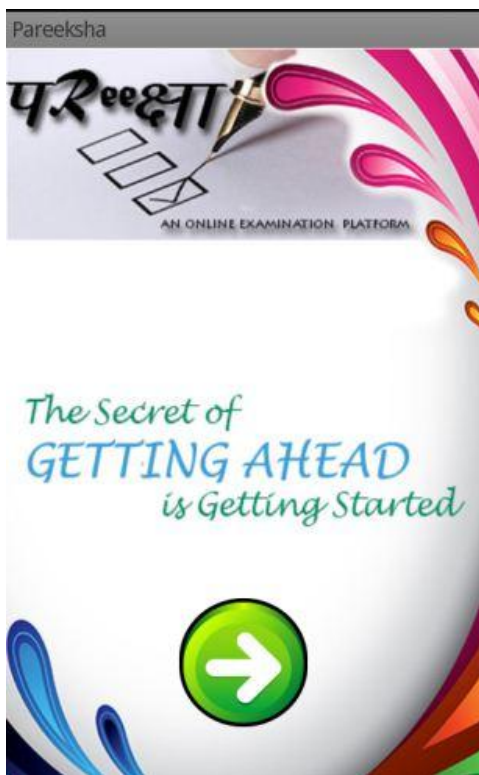
Fig. 7.2:   Start Screen

Fig. 7.3:   Home Screen

Fig. 7.4: Teacher SignUp

Fig. 7.5: Teacher Login

Fig. 7.6: Teacher Dashboard   **Fig. 7.7: Add Question**
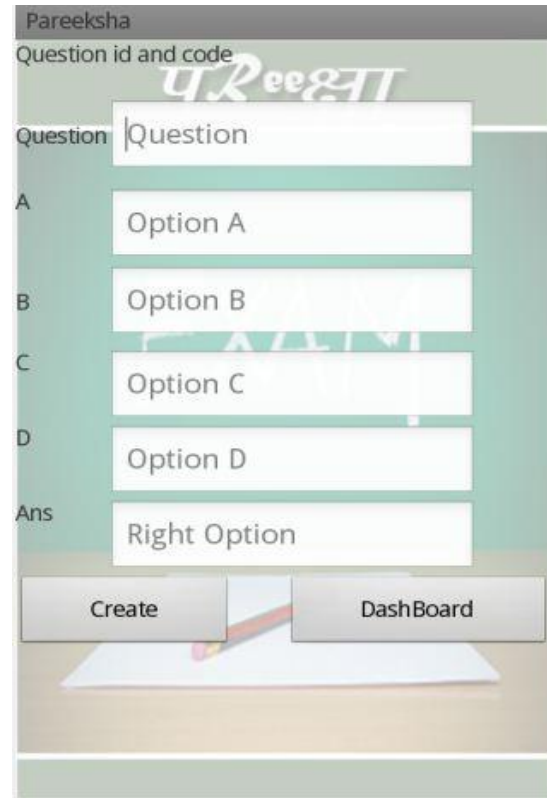
Fig. 7.8: Student SignUp

Fig. 7.9: Student Login
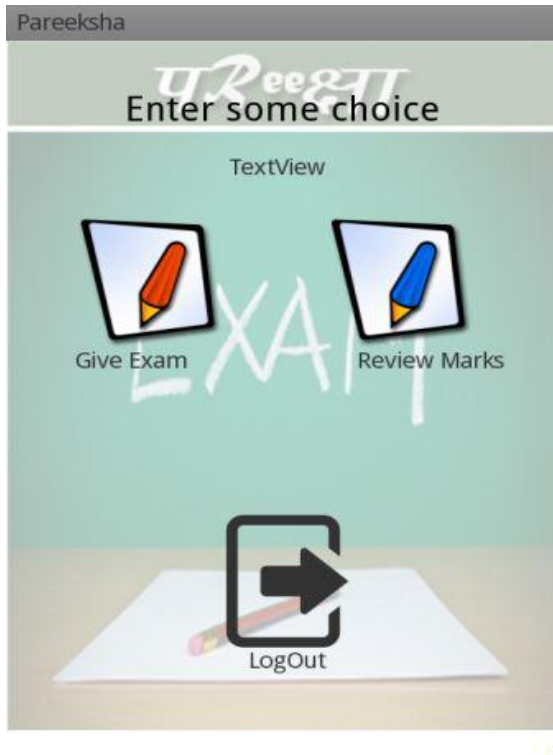
Fig. 7.10: Student Dashboard
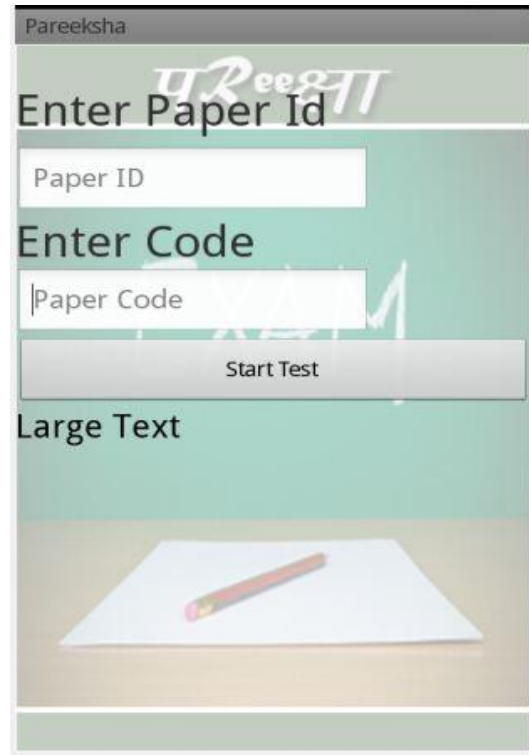
Fig. 7.11: Enter Test ID

Fig. 7.12: Exam Finished

7.4 Platform Used

7.4.1 Eclipse:

In computer programming, Eclipse is an integrated development environment (IDE). It contains a base workspace and an extensible plug-in system for customizing the environment. Written mostly in Java, Eclipse can be used to develop applications. By means of various plug-ins, Eclipse may also be used to develop applications in other programming languages: Ada, ABAP,C, C++, COBOL, Fortran, Haskell, JavaScript, Lasso, Natural, Perl, PHP, Prolog, Python, R, Ruby, Scala, Clojure, Groovy, Scheme, and Erlang. It can also be used to develop packages for the software Mathematica. Development environments include the Eclipse Java development tools (JDT) for Java and Scale, Eclipse CDT for C/C++ and Eclipse PDT for PHP, among others.

The initial codebase originated from IBM Visual Age. The Eclipse software development kit (SDK), which includes the Java development tools, is meant for Java developers. Users can extend its abilities by installing plug-ins written for the Eclipse Platform, such as development toolkits for other programming languages, and can write and contribute their own plug-in modules.

Released under the terms of the Eclipse Public License, Eclipse SDK is free and open source software (although it is incompatible with the GNU General Public License).

Eclipse uses plug-ins to provide all the functionality within and on top of the runtime system. Its runtime system is based on Equinox, an implementation of the OSGi core framework specification.

In addition to allowing the Eclipse Platform to be extended using other programming languages such as C and Python, the plug-in framework allows the Eclipse Platform to work with typesetting languages like LaTeX, networking applications such as telnet and database management systems. The plug-in architecture supports writing any desired extension to the environment, such as for configuration management. Java and CVS support is provided in the Eclipse SDK, with support for other systems provided by third-party plug-ins.

With the exception of a small run-time kernel, everything in Eclipse is a plug-in. This means that every plug-in developed integrates with Eclipse in exactly the same way as other plug-ins; in this respect, all features are "created equal". Eclipse provides plug-ins for a wide variety of

features, some of which are through third parties using both free and commercial models. Examples of plug-ins include for UML, for Sequence and other UML diagrams, a plug-in for DB Explorer, and many others.

The Eclipse SDK includes the Eclipse Java development tools (JDT), offering an IDE with a built-in incremental Java compiler and a full model of the Java source files. This allows for advanced refactoring techniques and code analysis. The IDE also makes use of a workspace, in this case a set of metadata over a flat filespace allowing external file modifications as long as the corresponding workspace "resource" is refreshed afterwards.

Eclipse implements widgets through a Java toolkit called SWT, whereas most Java applications use the Java standard Abstract Window Toolkit (AWT) or Swing. Eclipse's user interface also uses an intermediate graphical user interface layer called JFace, which simplifies the construction of applications based on SWT.

Language packs being developed by the "Babel project" provide translations into over a dozen natural languages.

Eclipse provides the Rich Client Platform (RCP) for developing general purpose applications. The following components constitute the rich client platform:

- Equinox OSGi – a standard bundling framework.
- Core platform – boot Eclipse, run plug-ins.
- Standard Widget Toolkit (SWT) – a portable widget toolkit.
- JFace – viewer classes to bring model view controller programming to SWT, file buffers, text handling, text editors.
- Eclipse Workbench – views, editors, perspectives, wizards.

Examples of rich client applications based on Eclipse are:

- IBM Notes 8 and 9
- Novell/NetIQ Designer for Identity Manager
- Apache Directory Studio
- Remote Component Environment

7.4.2 Virtual Router

Virtual Router uses the Virtual Router Redundancy Protocol .The Virtual Router Redundancy Protocol (VRRP) is a computer networking protocol that provides for automatic assignment of available Internet Protocol (IP) routers to participating hosts. This increases the availability and reliability of routing paths via automatic default gateway selections on an IP sub network.

The protocol achieves this by creation of virtual routers, which are an abstract representation of multiple routers, i.e. master and backup routers, acting as a group. The default gateway of a participating host is assigned to the virtual router instead of a physical router. If the physical router that is routing packets on behalf of the virtual router fails, another physical router is selected to automatically replace it. The physical router that is forwarding packets at any given time is called the master router.

VRRP provides information on the state of a router, not the routes processed and exchanged by that router. Each VRRP instance is limited, in scope, to a single subnet. It does not advertise IP routes beyond that subnet or affect the routing table in any way. VRRP can be used in Ethernet, MPLS and token ring networks with Internet Protocol Version 4(IPv4), as well as IPv6.

The protocol is described in IETF publication RFC 5798, which is an open standard, but Cisco claims that a similar protocol with essentially the same facility is patented and licensed; however in reply to a direct request Robert Barr of Cisco replied in 2001 that they will not assert any patent claims unless someone tried to assert a claim against Cisco. IBM also claims covering patents and their statement is readable on the IETF webpage

CHAPTER 8

TESTING

8. <u>TESTING</u>

This chapter includes strategies used for testing of the software. Testing is a set of activities that can be planned in advance and conducted systematically. It begins with testing of each module separately (Unit Testing). Unit testing refers to tests that verify the functionality of a specific section of code, usually at the function level. Then integration of modules has done. After that integration, testing of whole software uncovers overall errors of the software. Software testing is often referred to as verification and validation.

8.1 Software Testing

A strategy for software testing viewed in the context of the spiral. Unit testing begins at the vortex of the spiral and concentrates on each unit (i.e., component) of the software as implemented in source code. Testing progresses by moving outward along the spiral to integration testing, where the focus is on design and the construction of the software architecture. Taking another turn outward on the spiral, we encounter validation testing, where requirements established as part of software requirements analysis are validated against the software that has been constructed. Finally, we arrive at system testing, where the software and other system elements are tested as a whole. To test computer software, we spiral out along streamlines that broaden the scope of testing with each turn.

Software testing is also referred to as verification and validation. Verification refers to the set of activities that ensure that software correctly implements a specific function. Validation refers to a different set of activities that ensure that the software that has been built is traceable to customer requirements.
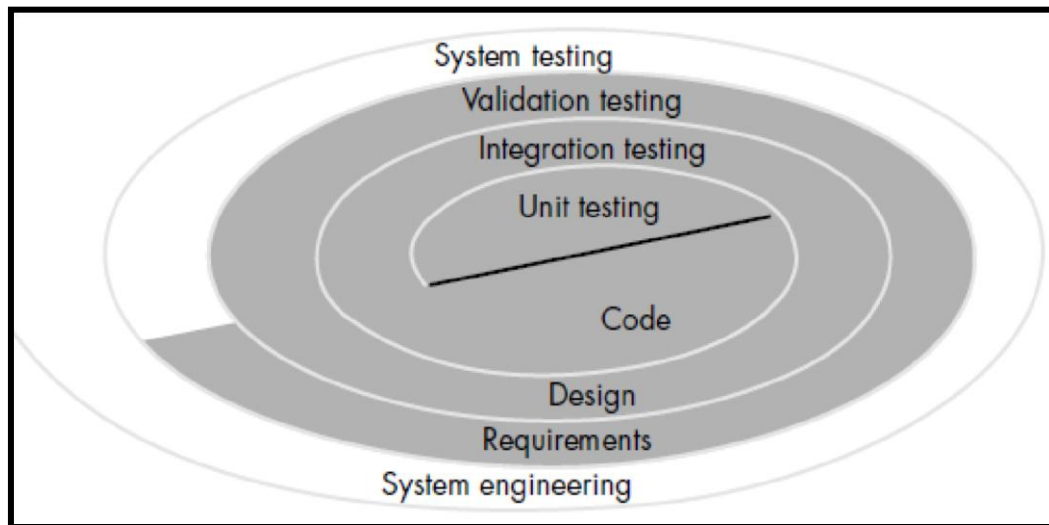
Fig 8.1 Testing

8.2. Unit Testing

Unit testing focuses verification effort on the smallest unit of software design—the software component or module. Using the component-level design description as a guide, important control paths are tested to uncover errors within the boundary of the module. The relative complexity of tests and uncovered errors is limited by the constrained scope established for unit testing. Testing has been done in various parts for each main module:

- Testing for representation module: The testing was done by plotting various data in existing software and comparing it to the data plotted in the software developed.

- Testing for Graphical Analysis module: In this module testing was done by plotting FFT of different data in existing software of selected channel and comparing it to the FFT generated by the software developed.

- Testing Image Processing module: The testing was done by generating tri map and single maps in existing software and comparing it to the maps generated by the software.

8.3. Integration Testing

Integration testing is a systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with interfacing. The objective is to take unit tested components and build a program structure that has been dictated by design.

Bottom-up Integration Testing

Bottom-up integration testing, begins construction and testing with atomic modules (i.e., components at the lowest levels in the program structure). Because components are integrated from the bottom up, processing required for components subordinate to a given level is always available.

A bottom-up integration strategy may be implemented with the following steps:

- Low level components are combined into clusters (sometimes called builds) that perform a specific software sub function.
- A control program for testing is written to coordinate test case input and output.
- The cluster is tested.
- Controls are removed and clusters are combined moving upward in the program structure

Using the Bottom up approach the software has been integrated with the help of several control flags. Modules are integrated by moving downward through the control hierarchy. The testing of the software is also done by comparing the FFT values of the software to the standard EEG software.

CHAPTER 9

RESULTS

9. Results

As we run our application on smart-phone with android version 2.2.3 (Gingerbread). We found out the following results in various steps

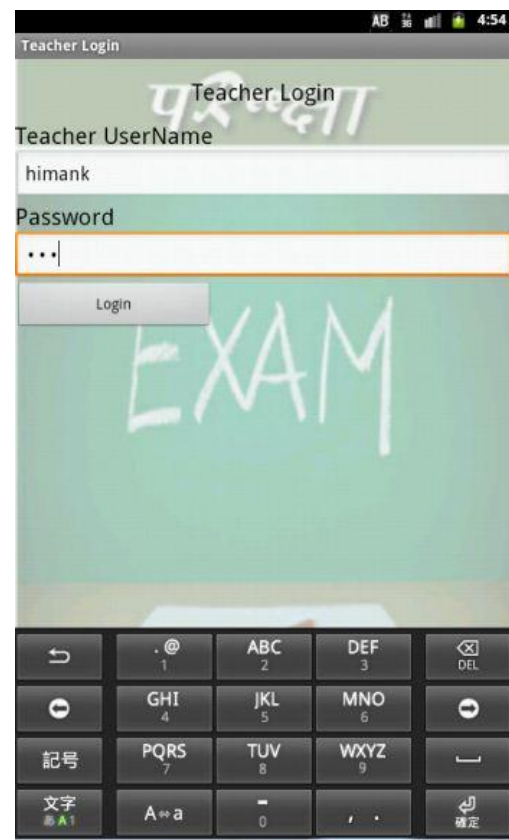**Fig. 9.1: Selecting mode**   **Fig. 9.2: Wi-Fi login**
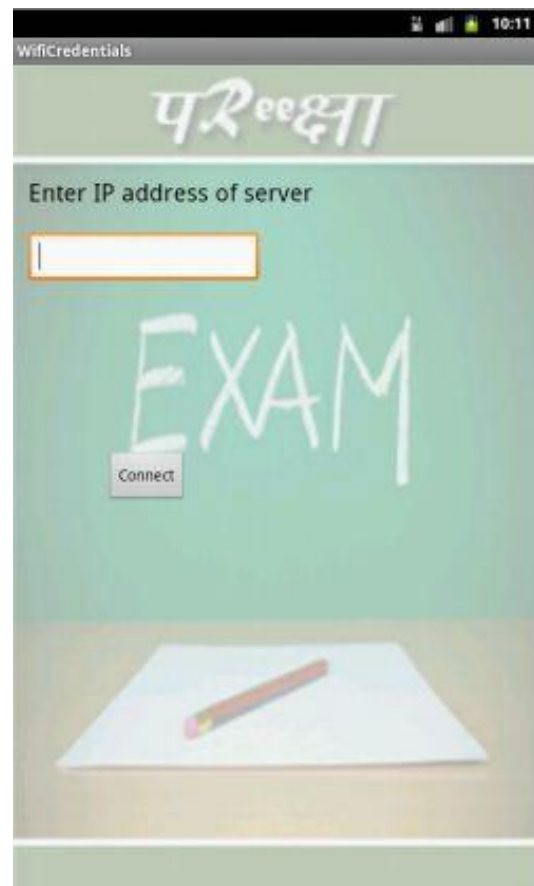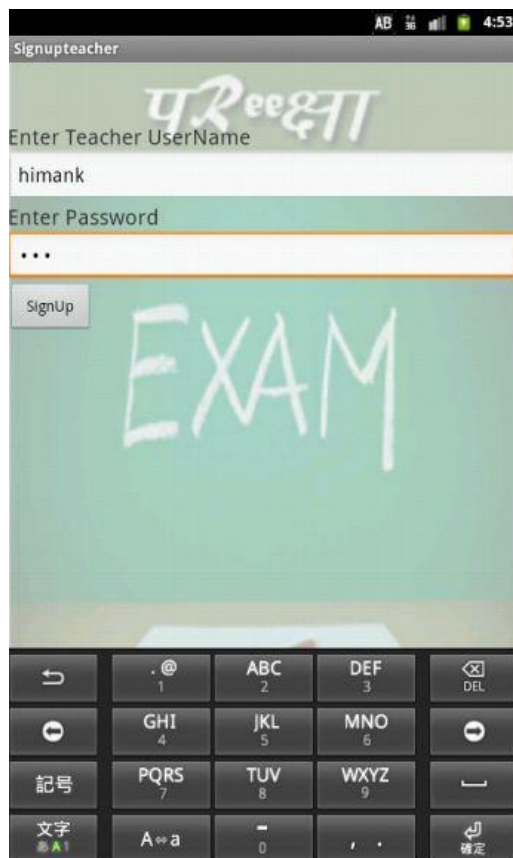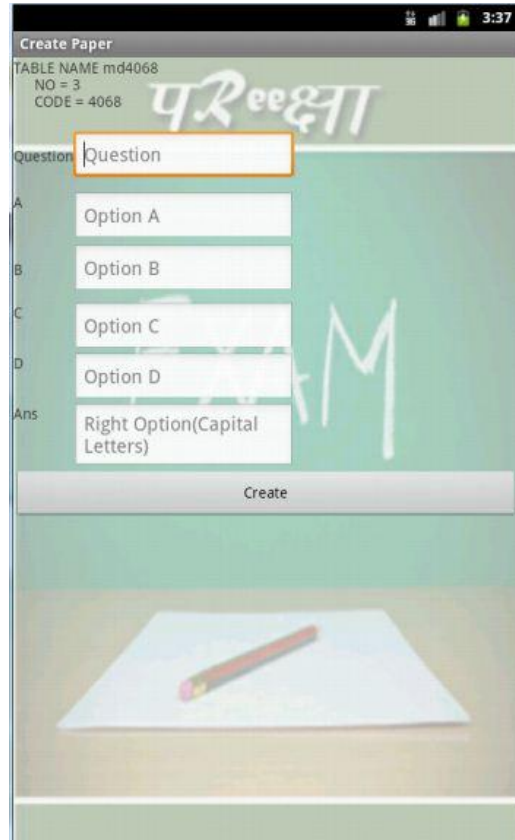
Fig. 9.3: Teacher Signup

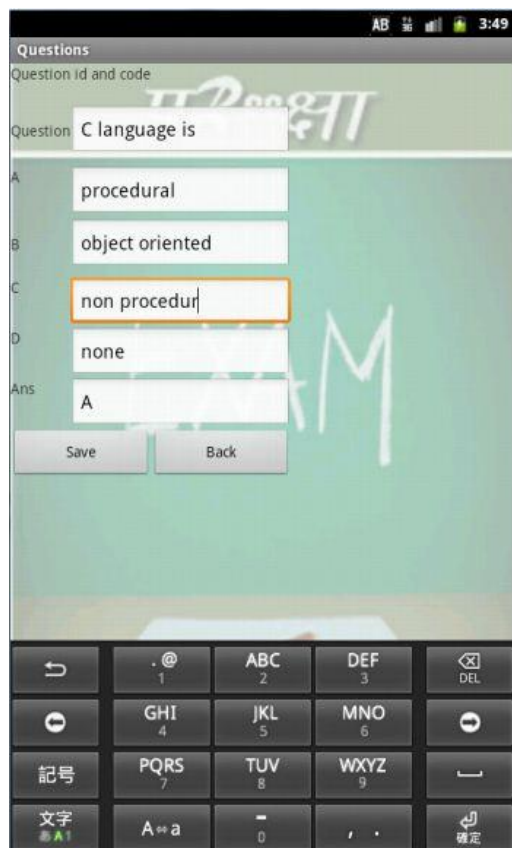**Fig. 9.4: Teacher Login**

Fig. 9.5: Teacher Dashboard

Fig. 9.6: Create Question Paper

Fig. 9.7: Adding Questions

Fig. 9.8: Adding Questions

Fig. 9.9: Edit Question

Fig. 9.10: Edit Question

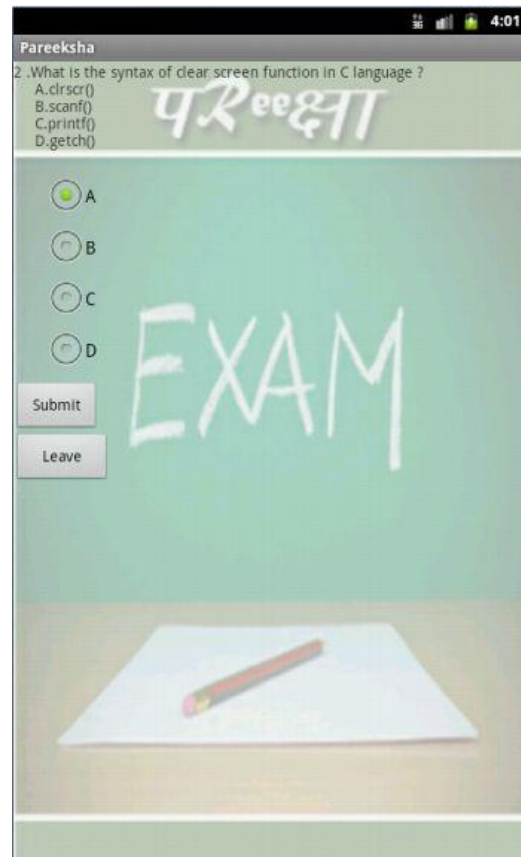**Fig. 9.11: Delete Question**   **Fig. 9.12: View Details**
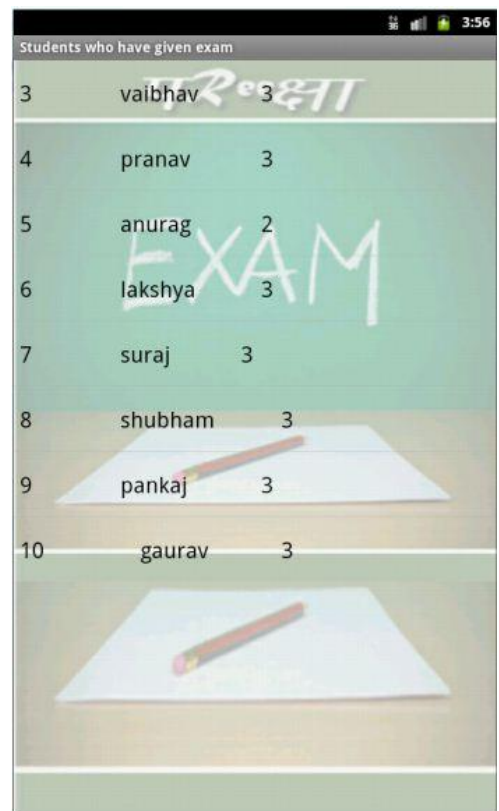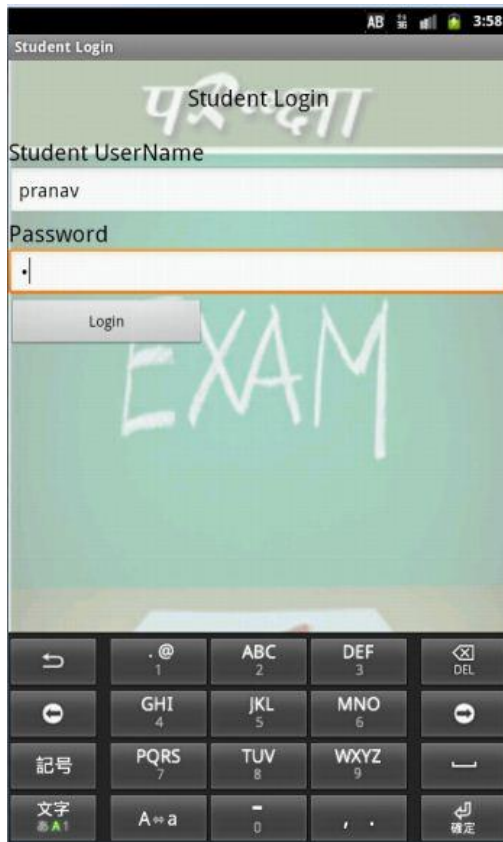
Fig. 9.13: Student Login **Fig. 9.14: Particular Exam**
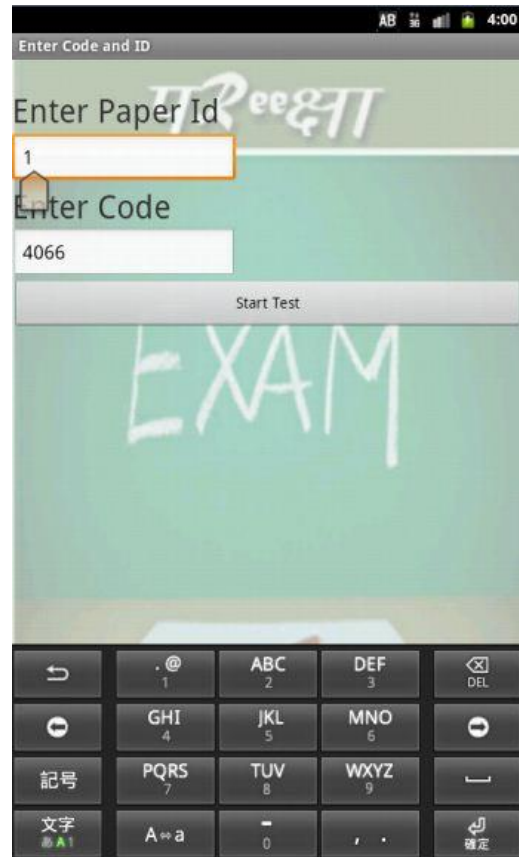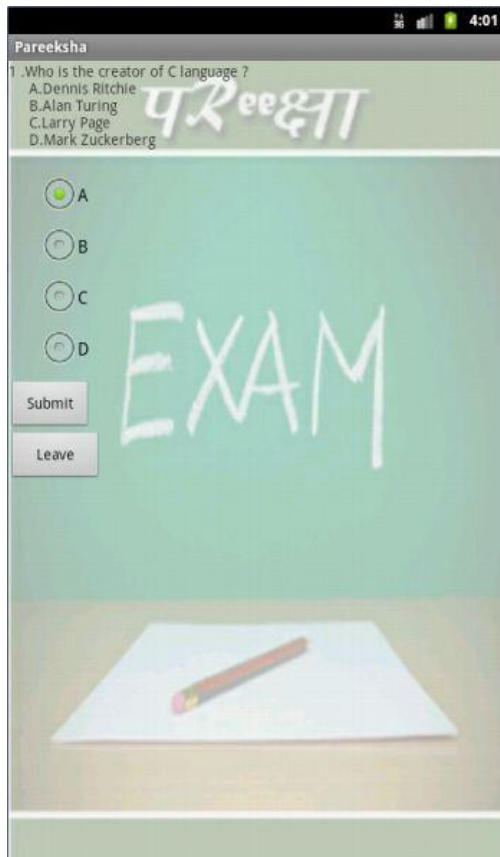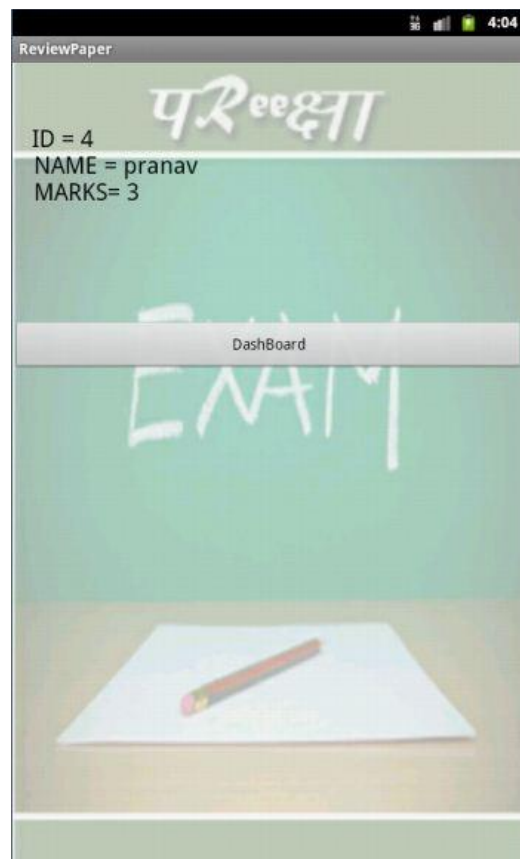
Fig. 9.15: Giving Exam   **Fig. 9.16: Giving Exam**
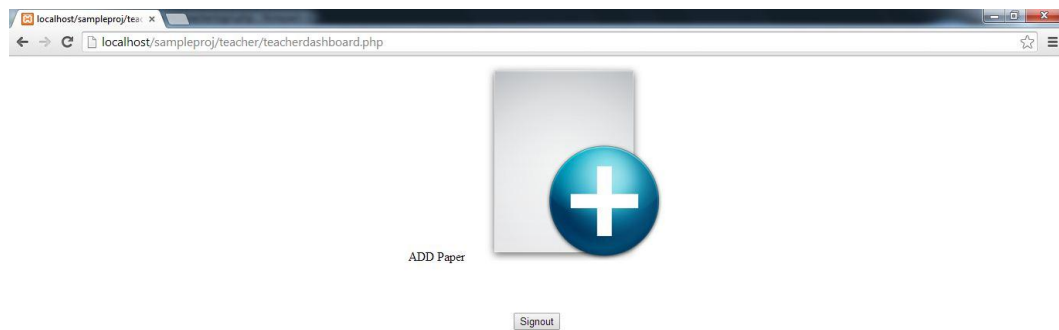
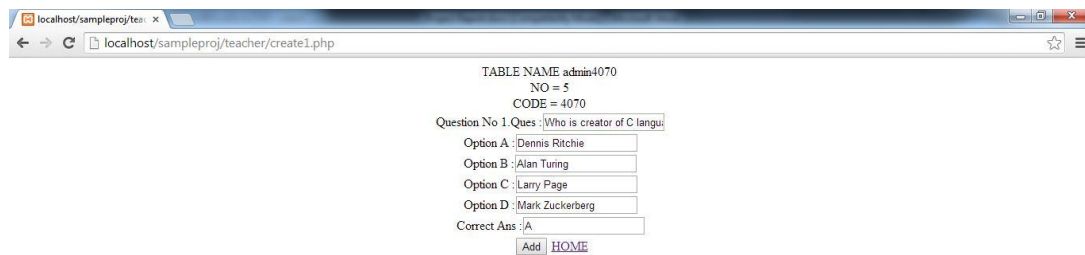**Fig. 9.17: Exam Finished**   **Fig. 9.18: Review marks**

Fig. 9.19 :   Web Login Page for Teacher

Fig. 9.20 : Create paper and add Questions

Fig. 9.21 : Adding Question through web

CHAPTER 10

CONCLUSION

10. Conclusion

In this fast changing technological era, education should be benefited by technology and ideas, and these two should step up coherently.

So in this Application we have tried to merge this simple idea of short quiz with this Android technology. It has been just 6 years that android has occupied the global smart-phone business and there are lots of opportunities for its development in future. On developing this application the most important component of this application is that we have successfully developed this application using Wi-Fi connectivity to the local server. This gives a great ease to the people who can't afford the web connectivity.

This application is still in growing phase as we have tested it successfully among 40 students in our hostel. We aspire to conduct the examination among all the university students. Here in present context we just developed the problem statement that we thought in initial phase i.e. to conduct the quick exam among more than 30 students on Android platform. In the future we intent to implement these following points in our android application:

1. Due to the modified library files included in each version of Android Operating System, we found it difficult to build the application for each version of Android OS. But we are trying to develop it for advance versions of Android OS.
2. Adding security measures like encryption to data in our Android application.
3. Adding automatic attendance system as component of our Android application.
4. Develop this application on window platform also.
5. Deploying this application worldwide through Google Play Store.

11. Literature Cited

1. Vogel, Lars (2013), Android ListView - Tutorial available on:
   http://www.vogella.com/articles/Android/article.html

2. MySql developer documentation available on:
   http://dev.mysql.com/tech-resources/articles/mysql_intro.html

3. W3Schools MySql tutorials available on:
   http://www.w3schools.com/php/php_mysql_intro.asp

4. Jonathan Strickland (2011), How the Google Phone Works available on:
   http://electronics.howstuffworks.com/google-phone2.htm

5. Android architecture reference available on:
   http://elinux.org/Android_Architecture

6. Android characteristics available on:
   .http://developer.android.com/about/dashboards/index.html

7. Brian Wilson (2005),‖ HTML Overview‖ available on:
   http://www.blooberry.com/indexdot/history/html.htm

8. Introduction to HTML available on:
   https://developer.mozilla.org/en-US/docs/Web/Guide/HTML/Introduction

9. Masonry documentation available on:
   http://masonry.desandro.com/docs/intro.html

10. Developer's documentation Google play services available on:
    http://developer.android.com/google/play-services/index.html

11. Android architecture design available on:
    www.android-app-market.com

12. Software testing available on:
    http://en.wikipedia.org/wiki/Software_testing

13. Google play services(Android market) available on:
    http://developer.android.com

APPENDIX A

Creating Android Application Development Environment

1. Download and install JDK 1.6 or higher

2. Set the path environment variable

3. Download and install the Android SDK

4. Update the path variable for completing installation of SDK tools

5. Install the ADT plugin for Eclipse (if you'll use the Eclipse IDE)

6. Download the latest SDK tools and platforms using the SDK Manager

APPENDIX B

B.1 Android Application: Steps for editing and using Android Module of the project

1. Copy the files in the CD to the desktop and open the project in eclipse.
2. Edit database connection to the web server and run the project.
3. The android application runs on android version 2.3.3 platform and needs internet for proper working so the connection wouldn't work on the android emulator
4. After downloading the application from the web site it can run on the android platform
5. After installing the application the application requires login username and password
6. If there is no username and password for the user then the user needs to create a new account as a student or as a teacher.
7. Use the application create profile, upload question paper etc.

B.2 Web Backend: Steps for editing and using backend module of the project

1. Copy the files from the CD drive to the desktop folder
2. Open project in Notepad++.
3. Import database in a ―pareeksha.sql database file in MySQL.
4. Start the xampp server and open apache server.
5. Use connectivity to data base as
   - Username: ―admin
   - Password: ―w
6. Web backend open in the browser and the android app is now connected to the server.