

# **MINI PROJECT**

**(2020-21)**

**Android-based Quiz Application**

**“Quiz Master”**

**PROJECT REPORT**



**Institute of Engineering & Technology**

**Submitted by –**

Rohan Bhardwaj (181500589)

Shashank Kumar Pandey (181500654)

Shivani Chauhan (181500678)

Vanshika Mahle (181500781)

Vimal Mishra (181500792)

**Supervised by –**

**Mr. Anand Parkash Gupta**

**Technical Trainer**

**Department of Computer Engineering & Application**



**Department of Computer Engineering and Applications**

**GLA University, Mathura**

**17 km. Stone NH#2, Mathura-Delhi Road, P.O. Chaumuhan, Mathura - 281406**

---

## **DECLARATION**

I hereby declare that the work which is being presented in the Mini Project “**Quiz Master**”, in partial fulfillment of the requirements for Mini-project Lab is an authentic record of my own work carried under the supervision of **Mr. Anand Parkash Gupta**, Technical Trainer.

Rohan Bhardwaj

Shashank Kr. Pandey

Shivani Chauhan

Vanshika Mahle

Vimal Mishra



**Department of Computer Engineering and Applications**

**GLA University, Mathura**

**17 km. Stone NH#2, Mathura-Delhi Road, P.O. Chaumuhan, Mathura - 281406**

---

## **CERTIFICATE**

This is to certify that the project entitled “**Quiz Master**”, carried out in Mini Project – I Lab, is a bonafide work by Rohan Bhardwaj (181500589), Shashank Kumar Pandey (181500654), Shivani Chauhan (181500678), Vanshika Mahle (181500781), and Vimal Mishra (181500792) and is submitted in partial fulfillment of the requirements for the award of the degree Bachelor of Technology (Computer Science & Engineering).

**Signature of Supervisor:**

**Name of Supervisor:** Mr. Anand Parkash Gupta

**Date:** 21/11/202

## ACKNOWLEDGEMENT

It gives us a great sense of pleasure to present the report of the B. Tech Mini Project undertaken during B. Tech. (Third Year). This project in itself is an acknowledgment of the inspiration, drive and technical assistance contributed to it by many individuals. This project would never have seen the light of the day without the help and guidance that we have received.

Our heartiest thanks to **Dr. (Prof). Anand Singh Jalal**, Head of Dept., Department of CEA for providing us with an encouraging platform to develop this project, which thus helped us in shaping our abilities towards a constructive goal.

We owe a special debt of gratitude to **Mr. Anand Parkash Gupta**, Technical Trainer, for his constant support and guidance throughout the course of our work. His sincerity, thoroughness, and perseverance have been a constant source of inspiration for us. He has showered us with all his extensively experienced ideas and insightful comments at virtually all stages of the project & has also taught us about the latest industry-oriented technologies.

We also would not like to miss the opportunity to acknowledge the contribution of all faculty members of the department for their kind guidance and cooperation during the development of our project. Last but not the least, we acknowledge our friends for their contribution to the completion of the project.

**Rohan Bhardwaj**

**Shashank Kr. Pandey**

**Shivani Chauhan**

**Vanshika Mahle**

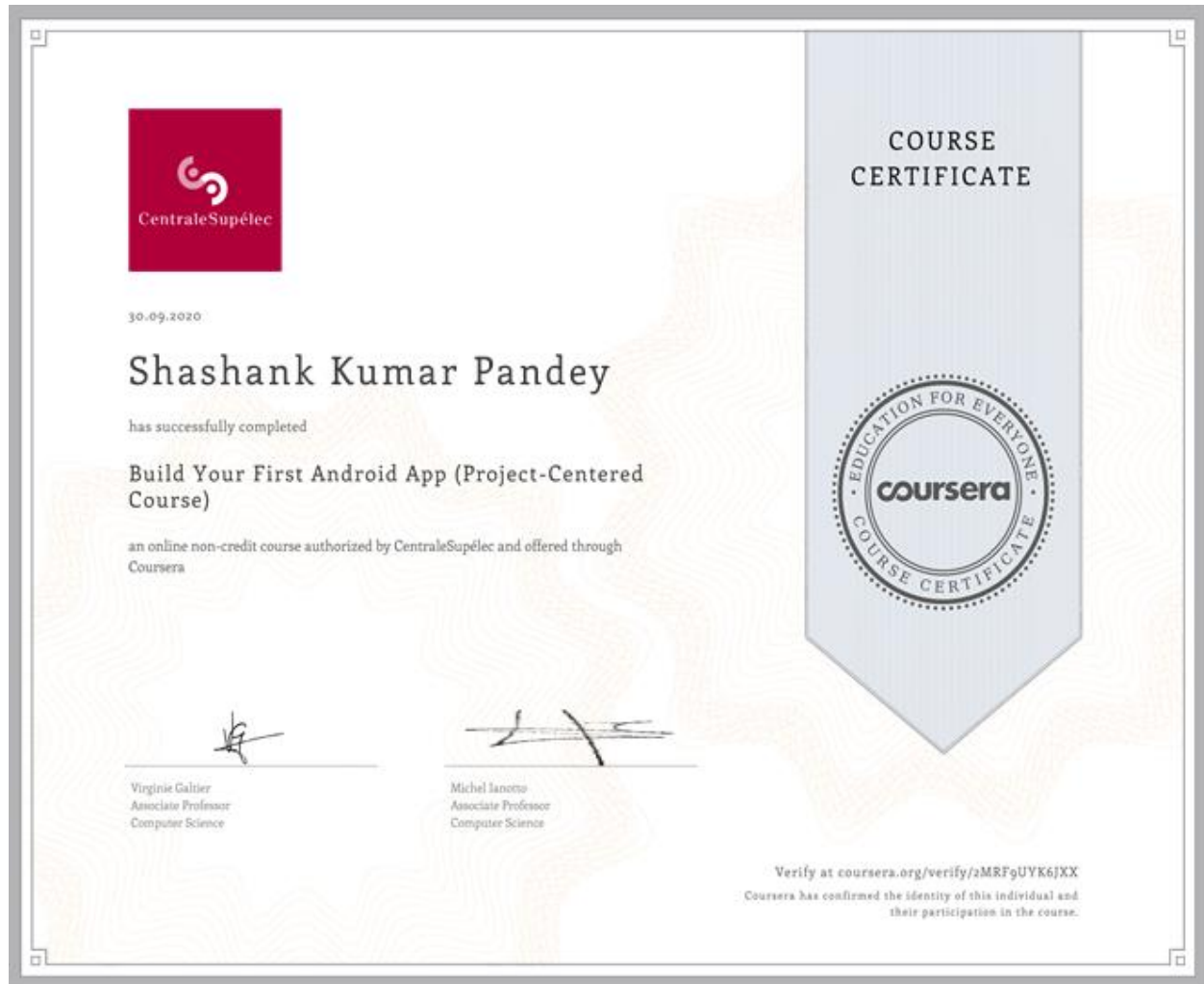
**Vimal Mishra**

# TRAINING CERTIFICATE

**Rohan Bhardwaj**



# Shashank Kumar Pandey



# Shivani Chauhan



# Vanshika Mahle





# Vimal Mishra



## **ABSTRACT**

As android is rapidly getting famous day by day and the number of its users is increasing with every blink of an eye because it is easy to access the necessary android based applications on smartphones and tablets in your hands. Therefore, we found this idea interesting, easy, and time-efficient to facilitate the users in this way without any difficulty. There are many online quiz applications available on the internet, but most of them are only for entertainment and fun. Moreover, if one is going to appear in any test, then it is difficult and time consuming for them to read the full books or articles related to specific fields for the preparation or revising their knowledge. But the most attractive feature of our app is that we take learning and fun side by side. Our app provides them the facility to revise their knowledge or to learn something advantageous in one place without wasting their time. This application deals with the development of an android-based multiple-choice question assessment system, namely: Quiz Master. The development of an android-based Quiz application is mainly required by students and learners to prepare themselves for different examinations directly through smartphones and tablets in their hands. One of the major goals of our project is to facilitate students in learning, gaining, and improving their knowledge & skills.

# TABLE OF CONTENTS

---

Declaration	i
Certificate	ii
Acknowledgement	iii
Training Certificate	iv
Abstract	ix
<b>1. Introduction</b>	<b>1</b>
1.1 Motivation and Overview	1
1.2 Objective	1
<b>2. Software Requirement Analysis</b>	<b>2</b>
2.1 Definition of Problem	2
2.2 Modules and their functionalities (SRS)	2
<b>3. System Architecture &amp; Design</b>	<b>8</b>
<b>4. Software Design</b>	<b>13</b>
4.1 Use Case Diagram	13
4.2 Block Diagram	14
4.3 Database Diagram	15
<b>5. Testing</b>	<b>16</b>
5.1 Unit Testing	16
5.2 Compatibility Testing	20
5.3 User Testing	20
<b>6. Implementation and User Interface</b>	<b>21</b>
<b>7. Conclusion</b>	<b>38</b>
<b>8. References</b>	<b>39</b>
<b>Appendices</b>	<b>40</b>

# **INTRODUCTION**

## **1.1 Motivation and Overview**

Online Learning has become the new normal. Smartphones are no longer a luxury. They have become a must-have as the COVID-19 pandemic has re-shaped education. Even during the toughest times, the world has never stopped learning.

Recent studies have shown that there is a drastic increase in the usage of technology by educational institutions to educate and assess the learnings of the students online.

It has been observed that Multiple Choice Questions (MCQs) based assessments are an effective way to evaluate one's learnings. It motivates us to develop a 'want' to learn more and remember what we have learned. An Online Quiz Application is a better alternative to paper-based offline evaluation techniques. It is user-friendly and reduces manual effort, thereby saving time & resources.

It is an application where a user with the admin privilege can do administrative tasks like add, delete, and edit from the application user interface and the user can participate in the MCQ quiz with a time limit. So, this project facilitates users to configure quizzes as well as giving quizzes using smartphones, tablets, and laptops.

## **1.2 Objective**

This project proposes an idea to develop a user-friendly, android based application that can educate as well as entertain users with a quick quiz on any category. The application will contain multiple sets of questions in each category. There will be a specific time limit set for answering each question. After completion of the contest, scores will be displayed which will help the user to improve their performance. It provides a fun-loving way to learn, so it can be used by people of all age groups who want to learn and assess their knowledge.

# **SOFTWARE REQUIREMENT ANALYSIS**

## **2.1 Definition of Problem**

It is essentially required to assist students in the learning and preparation of different tests conducted. This is what we tried to address in the development of Quiz Master: Quiz Application Development using Android-Based Platform, which can provide candidates with the preparation of tests in both user-friendly and interactive way. We designed the application to facilitate the users to be able to take short quizzes using portable devices such as smartphones and tablets.

## **2.2 Modules and Functionalities (SRS)**

The technologies used in our project include:

- **Java Development Kit (JDK)**

The Java Development Kit (JDK) is an implementation of either one of the Java Platform, Standard Edition, Java Platform, Enterprise Edition, or Java Platform, Micro Edition platforms released by Oracle Corporation in the form of a binary product aimed at Java developers on Solaris, Linux, macOS or Windows. The JDK includes a private JVM and a few other resources to finish the development of a Java application. Since the introduction of the Java platform, it has been by far the most widely used Software Development Kit (SDK).

- **Android SDK**

The Android software development kit (SDK) includes a comprehensive set of development tools. These include a debugger, libraries, a handset emulator

based on QEMU, documentation, sample code, and tutorials. Currently supported development platforms include computers running Linux (any modern desktop Linux distribution), Mac OS X 10.5.8 or later, and Windows 7 or later. As of March 2015, the SDK is not available on Android itself, but software development is possible by using specialized Android applications.

Until around the end of 2014, the officially-supported integrated development environment (IDE) was Eclipse using the Android Development Tools (ADT) Plugin, though IntelliJ IDEA IDE (all editions) fully supports Android development out of the box, and NetBeans IDE also supports Android development via a plugin. As of 2015, Android Studio, made by Google and powered by IntelliJ, is the official IDE; however, developers are free to use others, but Google made it clear that ADT was officially deprecated since the end of 2015 to focus on Android Studio as the official Android IDE. Additionally, developers may use any text editor to edit Java and XML files, then use command-line tools (Java Development Kit and Apache Ant are required) to create, build and debug Android applications as well as control attached Android devices (e.g., triggering a reboot, installing software package(s) remotely).

Enhancements to Android's SDK go hand-in-hand with the overall Android platform development. The SDK also supports older versions of the Android platform in case developers wish to target their applications at older devices. Development tools are downloadable components, so after one has downloaded the latest version and platform, older platforms and tools can also be downloaded for compatibility testing.

Android applications are packaged in .apk format and stored under /data/app folder on the Android OS (the folder is accessible only to the root user for security reasons). APK package contains .dex files (compiled byte code files called Dalvik executables), resource files, etc.

- **Firebase Firestore**

Cloud Firestore is a flexible, scalable database for mobile, web, and server development from Firebase and Google Cloud Platform. Like Firebase Realtime Database, it keeps your data in sync across client apps through real time listeners and offers offline support for mobile and web so you can build responsive apps that work regardless of network latency or Internet connectivity. Cloud Firestore also offers seamless integration with other Firebase and Google Cloud Platform products, including Cloud Functions.

The Cloud Firestore data model supports flexible, hierarchical data structures. Store your data in documents, organized into collections. Documents can contain complex nested objects in addition to subcollections.

Cloud Firestore caches data that your app is actively using, so the app can write, read, listen to, and query data even if the device is offline. When the device comes back online, Cloud Firestore synchronizes any local changes back to Cloud Firestore.

It protects access to our data in Cloud Firestore with Firebase Authentication and Cloud Firestore Security Rules for Android, iOS, and JavaScript, or Identity and Access Management (IAM) for server-side languages.

## **API**

API stands for Application Programming Interface. Together with backend databases, APIs provide a set of fixed rules and specifications that define interactions between software components.

An API can be created for libraries, operating systems, and applications. A good API makes it easier and faster to develop an app by providing all of the “building blocks” for it.

APIs can also be a great marketing tool or can even be your core software product.

## **LAYOUT**

A layout defines the structure for a user interface in your app, such as in an activity. All elements in the layout are built using a hierarchy of View and ViewGroup objects. A View usually draws something the user can see and interact with. Whereas a ViewGroup is an invisible container that defines the layout structure for View and other ViewGroup objects.

The View objects are usually called "widgets" and can be one of many subclasses, such as Button or TextView. The ViewGroup objects are usually called "layouts" and can be one of many types that provide a different layout structure, such as LinearLayout or ConstraintLayout.



## **ACTIVITY**

The Activity class is a crucial component of an Android app, and the way activities are launched and put together is a fundamental part of the platform's application model. Unlike programming paradigms in which apps are launched with a main() method, the Android system initiates code in an Activity instance by invoking specific callback methods that correspond to specific stages of its lifecycle.

## **FUNCTIONALITIES OF QUIZ MASTER APP:**

- Choose from Multiple Categories
- Choose From Multiple Sets
- View Question Number
- Countdown Timer
- Choose From Multiple Options
- View Correct Answers
- Earn Points for Correct Answers
- Know Your Final Score
- Play Again to Improve

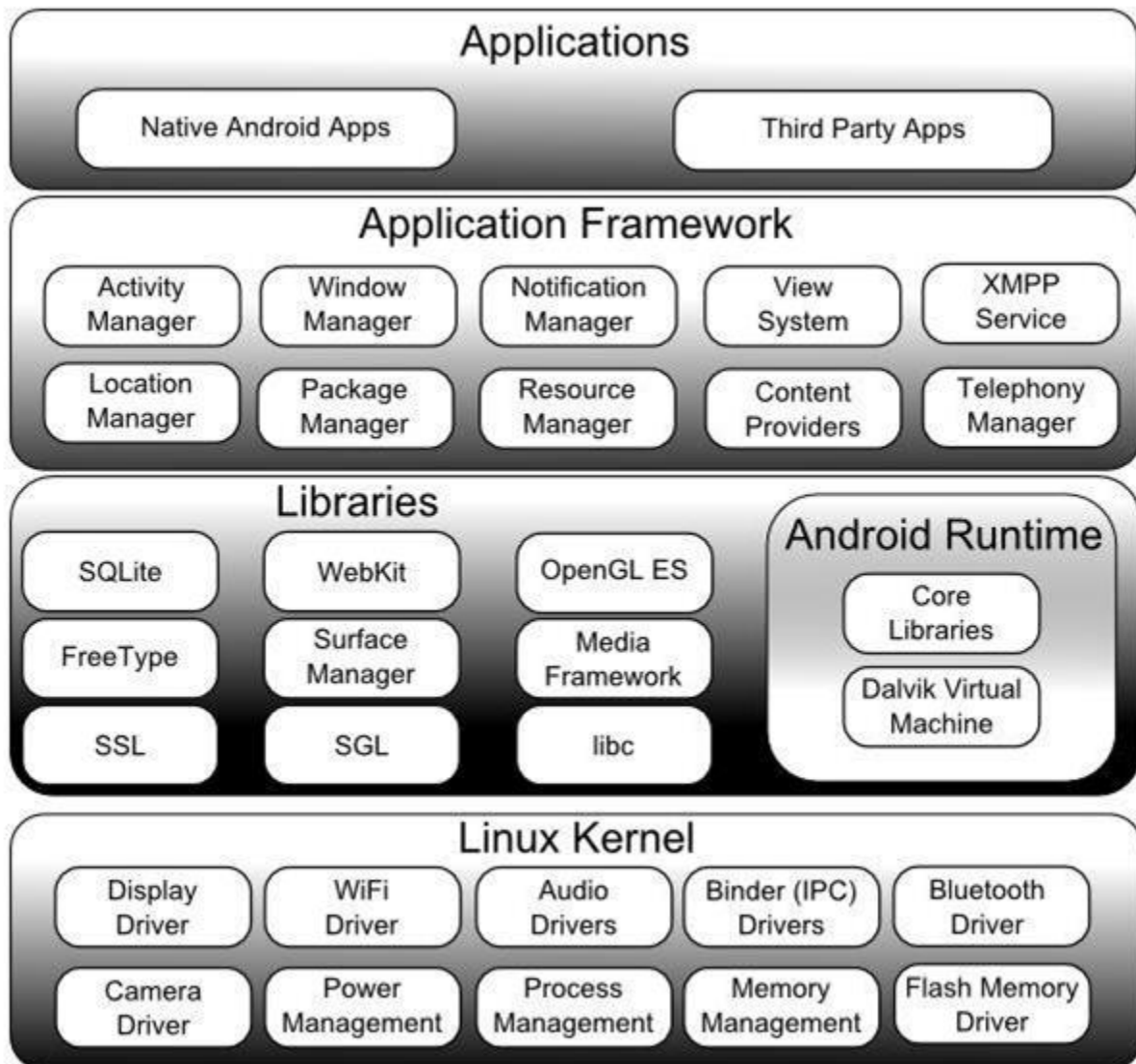
## **FUNCTIONALITIES OF QUIZ MASTER ADMIN APP:**

- Login for Registered User

- Add New Categories
- Delete Categories from The List
- Update Category Names
- Add New Sets in Each Category
- Delete Sets
- Add New Questions in the Sets
- Update Previous Questions
- Delete Questions

# SYSTEM ARCHITECTURE & DESIGN

## 3.1 Android Architecture



The different software components in the Android Operating System are arranged in the form of a stack. The different software components include

- The Linux Kernel

- Libraries
- Android Run Time
- Application Framework Applications

## **LINUX KERNEL:**

The Linux Kernel is the bottom layer of the architecture and the whole android operating system is built on the top of the Linux Kernel with some changes in the architecture. It is built on Linux 2.6 Kernel and some changes are made to its architecture. Linux Kernel is the core of the operating system. Process Management, Memory Management and Device Management are some of the functionality provided by the Linux Kernel.

## **ANDROID RUNTIME -DALVIK VIRTUAL MACHINE:**

Android Runtime comprises Dalvik Virtual Machine and Core Java Libraries.

The applications in android devices are run using Dalvik virtual machine which is a type of JVM. The application that is run consumes very less power and low memory. The instance in dalvik virtual machine enables every android application to run its own processes. .dex files are run by dalvik virtual machines like .class files in JVM, these .dex files are generated from .class files at compile time and these .dex files provide higher efficiency.

The virtual machine that is introduced by Google is the ART (Android Run Time). This ART was introduced in recent times it is introduced with Lollipop and this ART has completely replaced the Dalvik Virtual machine and has its own advantages over the Dalvik machine.

Android Core Java libraries are also part of the android runtime and these libraries provide most of the functionalities.

## **LIBRARIES:**

The Android Libraries are written in C/ C++ language and each device has its own library. The Important libraries include.

- Surface Manager
- SQLite
- WebKit
- OpenGL
- Media Framework
- SSL

**SSL** is a library that provides security on the internet.

**SQLite** is the library which is the database base engine and that is used to store and process the data.

**Webkit** is the browser library or engine that displays the HTML Content in the application.

**OpenGL** is used to bind the 2D/ 3D content to the screen.

**Media Framework** provides different codecs for playing/ recording different media types.

## **ANDROID APPLICATION FRAMEWORK:**

The Android applications directly interact with the Android Framework to run and are managed here. Resource Management, Voice call management and activities like these are handled by the application framework.

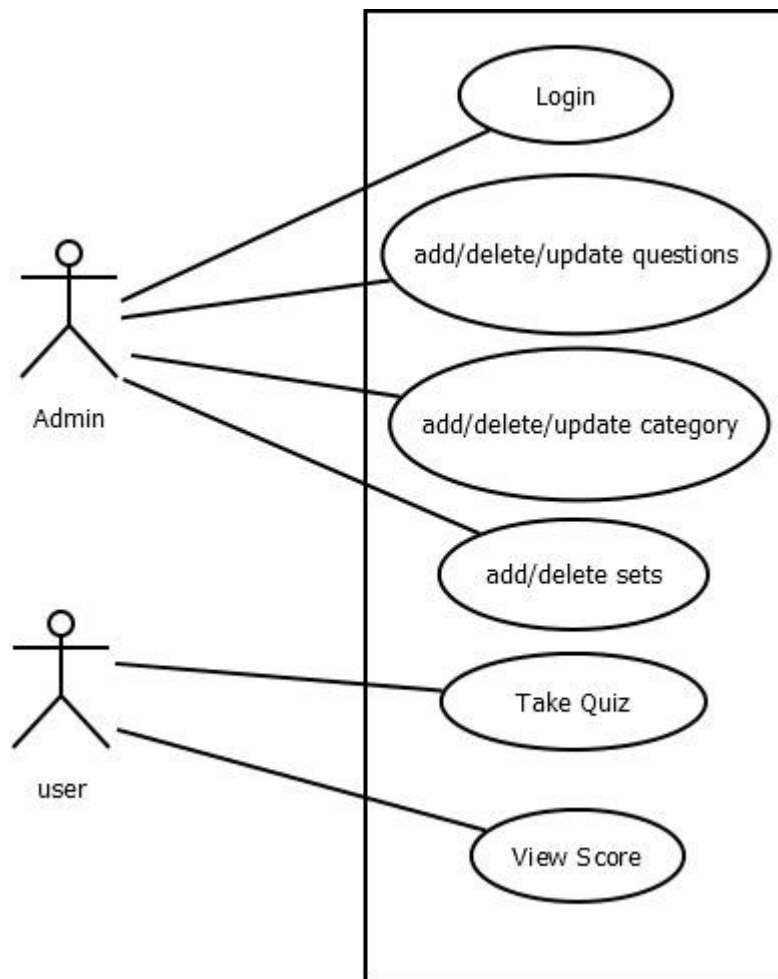
The Android framework includes the following key services:

- **Activity Manager** – The activity stack and the application lifecycle are controlled by the activity manager.
- **Content Providers** – The data is shared and published with other applications using this content provider.
- **Resource Manager** – The non-code embedded resources such as strings, color settings and user interface layouts are accessed using this resource manager.
- **Notifications Manager** – The display alerts and notifications to the user are given by this notifications manager.
- **View System** – The user interfaces of the application are created by using this view system.
- **Package Manager** – The other applications which are in the current device knows information about the other applications that are in the device using this package manager.

- Telephony Manager – The information such as status and subscriber is provided to the application using the telephony manager.
- Location Manager – The changes in the location and the updating of the location is done by this location manager.

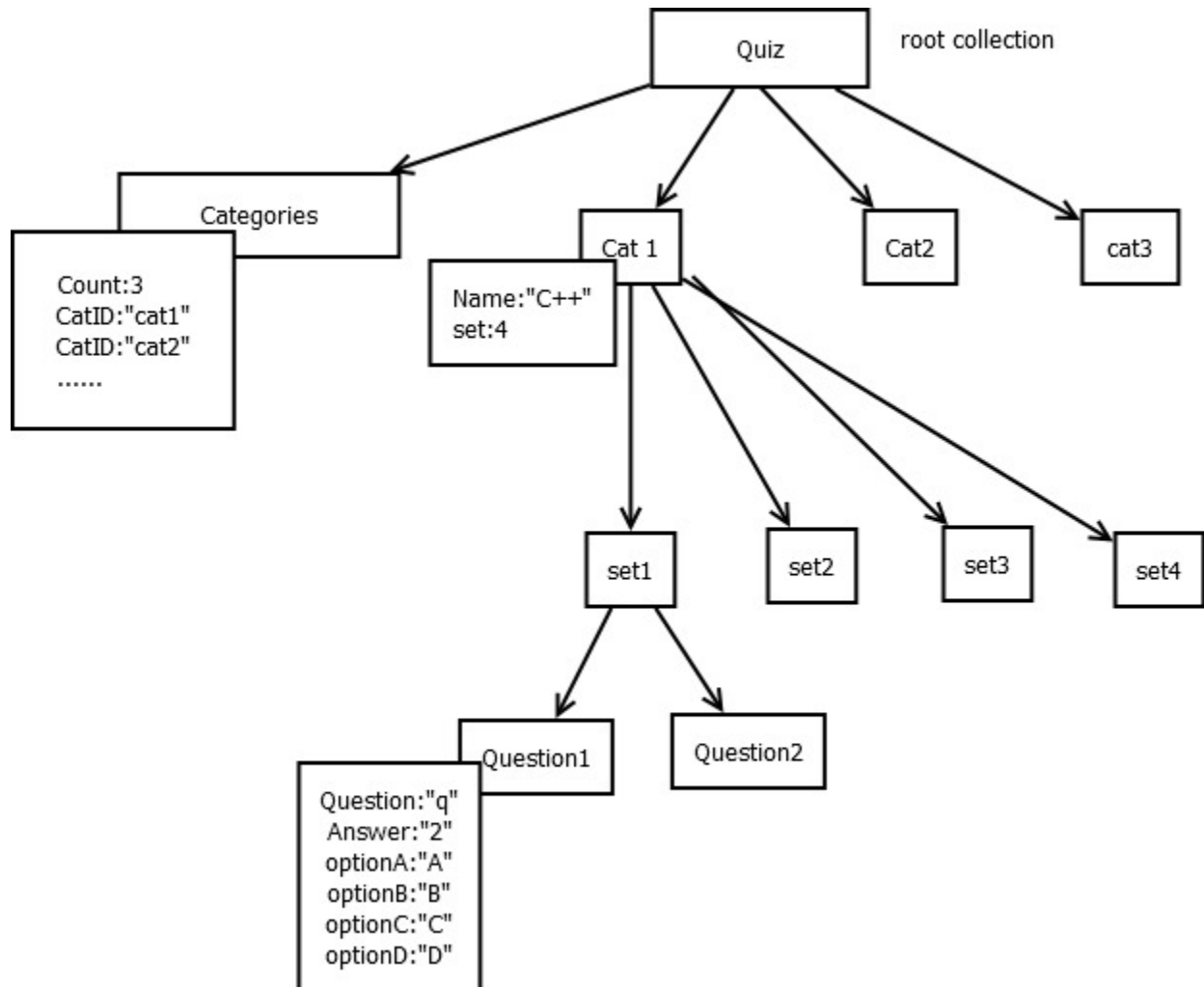
# SOFTWARE DESIGN

## 4.1 Use Case Diagram

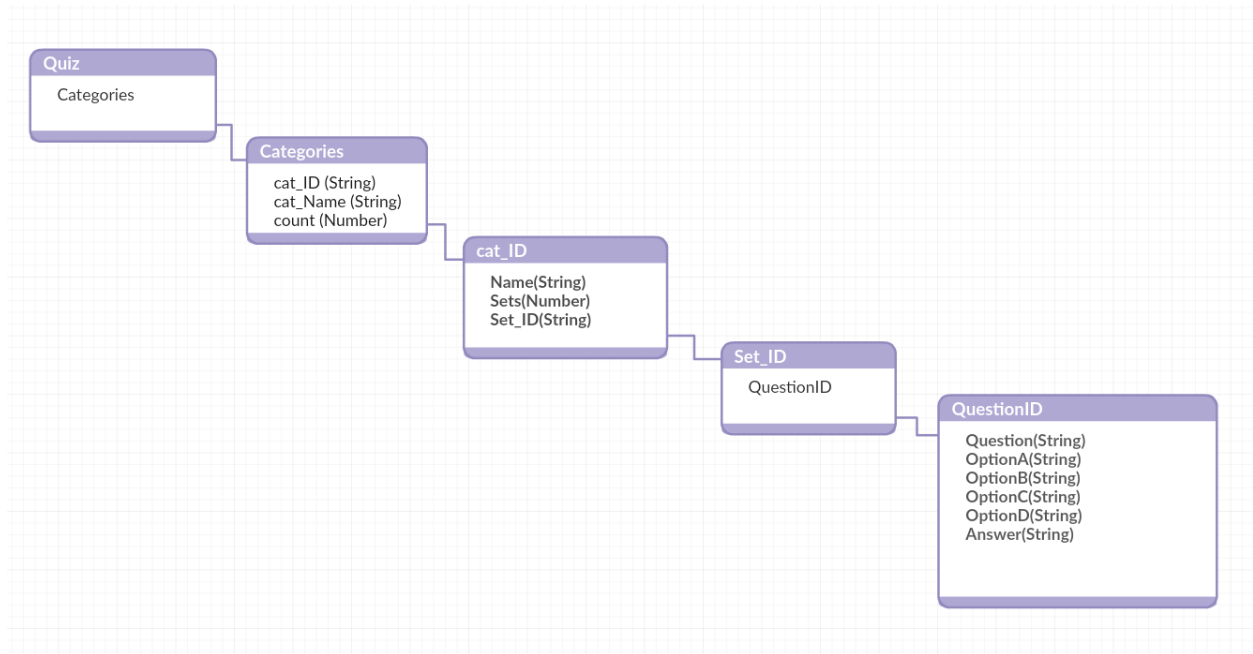




## 4.2 Block Diagram



## 4.3 Database Diagram



# TESTING

## 6.1 UNIT TESTING

Unit Testing is a software testing method in which small parts of the code are separated from other parts and the functionality is checked whether it works as expected or not.

**Table 1 Unit Testing – Client App**

Test Case	Expected Result	Pass/ Fail
Start Page – Launch Screen	Should display splash screen with animated text	Pass
Main Screen display	Should display start button	Pass
Categories Screen display	Should display multiple categories to choose from and move to sets activity on selecting any category	Pass

Sets Screen Display	Should display the number of sets available for the selected category	Pass
Selecting a set	Should display the question in the set and start the quiz	Pass
Question Screen Display	Should show the current question number and the countdown timer along with the questions	Pass
Selecting an option	Should change from white to green if answer is correct & red if the answer is incorrect	Pass
Score Activity Screen	Should Display the final score and the button to finish the quiz	Pass
Pressing the Finish Button	Should go back to the Main Screen	Pass

**Table 2 Unit Testing – Admin App**

Launch Screen display	Should display the Splash Screen with animated text	Pass
Login Screen Display	Should display login page for admin to enter email and password	Pass
Pressing Login Button	Should login to the app if correct id and password are provided, else print incorrect details entered	Pass
Category Screen Display	Should display a list of categories available in the database	Pass
Pressing Add Category Button	Should display a dialog box to enter the new category name to be added	Pass
Pressing the delete icon in Category screen	Should display a confirm dialog box to confirm deletion of category	Pass

Long pressing the category name	Should display a dialog box to enter the updated name for the category	Pass
Pressing add set button	Should add a set in the category	Pass
Pressing delete icon in set screen	Should display a confirm dialog to delete the selected set	Pass
Questions Screen display	Should display the total number of questions in the set	Pass
Pressing add questions button	Should display the screen to type new question along with the 4 options and correct answer	Pass
Pressing delete icon in questions screen	Should display a confirm dialog to delete the question selected	Pass
Clicking on the question number	Should display the screen to edit and update the question	Pass

Pressing back buttons	Should move to previous activity	Pass
-----------------------	----------------------------------	------

## 6.2 COMPATIBILITY TESTING

This application was tested and used on different devices like LG G3, Google Nexus 4. The application worked fine and is stable. The application worked fine in portrait mode and there isn't any problem with compatibility.

## 6.3 USER TESTING

The present application was tested by our classmates who are using different mobile devices that have Android Lollipop, Android Nougat, Android Oreo and that seemed to be working fine and they were satisfied with the performance and responsiveness of the application and how the app worked.

## **IMPLEMENTATION AND USER INTERFACE**

We have divided our project into two applications, one used by the user and one used by the admin.

### **Part 1:**

Step 1 – We have first build splash activity to start the application.

Step 2 – Then, we have the main activity which consists of a start button.

Step 3 – We made a category layout of our quiz app which includes various topics on which we want to attempt the quiz.

Step 4 – After it, we created a set activity that has a different number of sets.

Step 5 – Then, we have created the question activity layout of our quiz app which consists of questions with a countdown set on it. Also, it shows the correct option with green color and the incorrect chosen option with red color.

Step 6 – We have created a score activity of our quiz app which displays the total score gained by the user.

Step 7 – We have connected our android app to Google Firebase.

Step 8 – Then, we have created the Firestore Cloud Database that will be used in our quiz app.

Step 9 – After it, we have fetched Categories List from Firestore Database.

Step 10 – Then, we have fetched no. of sets in a category from Firestore



Database.

Step 11 – At last, we have fetched the questions list from the Firestore Database.

## **Part 2: (Admin App)**

Step 1 – We have first created a login activity for our admin app.

Step 2 – Then, we made a Category activity.

Step 3 – After it, we have added functionality for adding a new category to the app.

Step 4 – We added delete category functionality to our quiz admin app.

Step 5 – Then, added edit category functionality to our app.

Step 6 – We created sets of activity layouts for the admin app.

Step 7 – Then, we added new settings to the app.

Step 8 – We added features to delete sets.

Step 9 – After this, we created a questions activity for the app.

Step 10 – We added another feature to delete questions in our app.

Step 11 – Then, we added a feature to add new questions to the app.

Step 12 – We added a feature to edit questions in our quiz admin app.

Step 13 – After it, we secured our admin app database.

Let us look into the user interface.

## **CLIENT APP**





Are You Ready?

**LET'S BEGIN**



## Categories

**C++**

**Python**

**Java**

**CSS**



CSS

## Sets

1

2

3

4

5

6

7

**2/3**

**What is the maximum possible length of an identifier?**

**27**

**31 characters**

**63 characters**

**79 characters**

**None of the above**

**2/3**

**What is the maximum possible length of an identifier?**

**24**

**31 characters**

**63 characters**

**79 characters**

**None of the above**

**1/3**

**Is Python case sensitive when dealing with identifiers?**

**26**

**Yes**

**No**

**Machine dependent**

**None of the mentioned**



**Score**

**1/3**

**FINISH**

## QUIZ MASTER ADMIN APP



### Quiz Master Admin

Email ID

Password

**LOG IN**

## Categories

**C++**



**Python**



**Java**



**CSS**



**Add New Category**

**SET 1**



**SET 2**



**Add New Set**



## Questions

**QUESTION 1**



**QUESTION 2**



**QUESTION 3**



**Add New Question**



## Question 1

Is Python case sensitive when dealing with identifiers?

---

Yes

---

No

---

Machine dependent

---

None of the mentioned

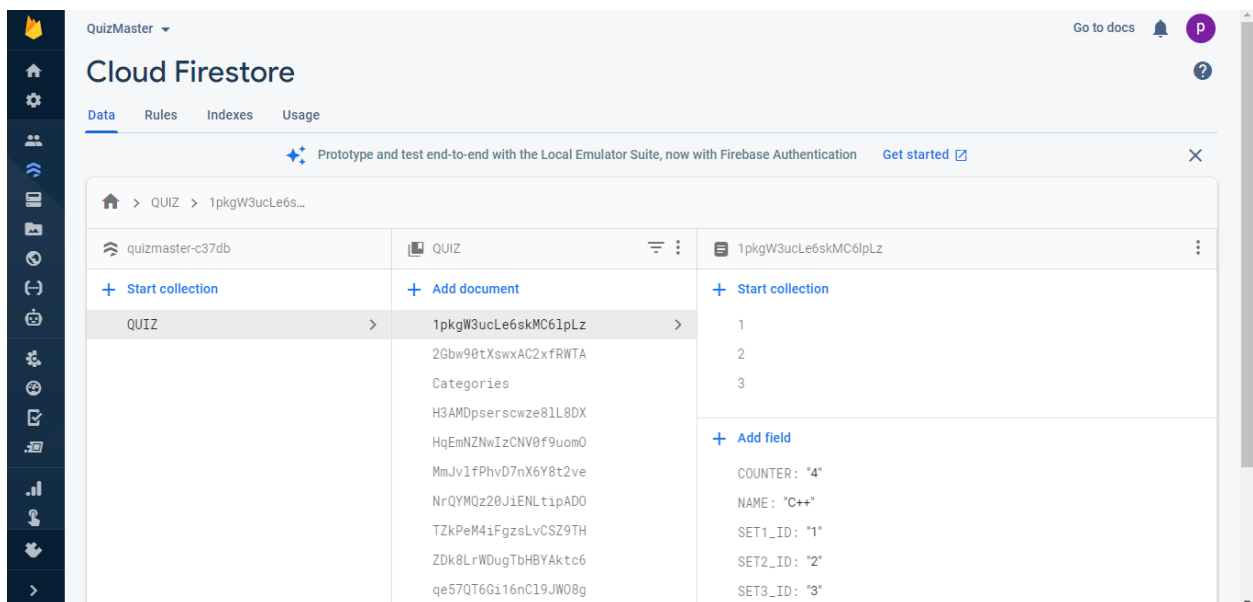
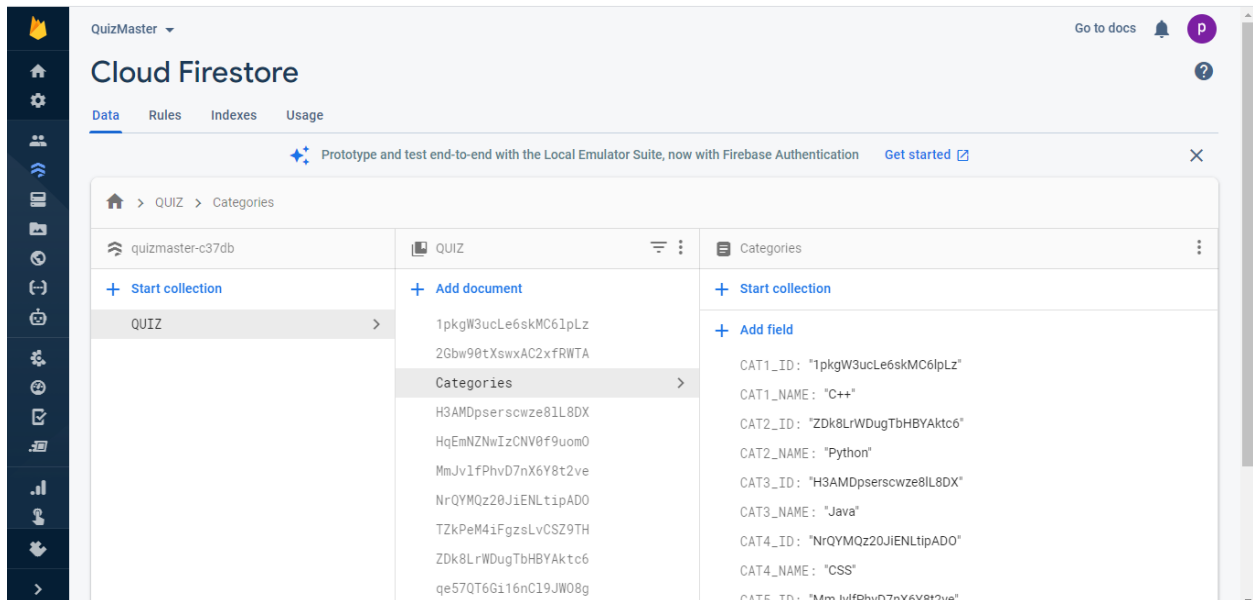
---

1

---

UPDATE

# FIREBASE FIRESTORE INTEGRATION



QuizMaster Go to docs ?

## Cloud Firestore

[Data](#) [Rules](#) [Indexes](#) [Usage](#)

[Prototype and test end-to-end with the Local Emulator Suite, now with Firebase Authentication](#) [Get started](#)

[QUIZ](#) > [1pkgW3ucLe6s...](#) > 1

QUIZ	1pkgW3ucLe6skMC6lpLz	1
<a href="#">+ Add document</a> 1pkgW3ucLe6skMC6lpLz 2GbW90tXswxAC2xfrWTA Categories H3AMDpersowze8lL8DX HqEmNZNIzCNV0f9uom0 MmJv1FPhvD7nX6Y8t2ve NrQYMQz20JiENLtipAD0 TZkPeM4iFgzLVCsz9TH ZDk8LrWDugTbHBYAkte6 qe57QT6G116nCl9JW08g	<a href="#">+ Start collection</a> 1 2 3 <a href="#">+ Add field</a> COUNTER: "4" NAME: "C++" SET1_ID: "1" SET2_ID: "2" SET3_ID: "3"	<a href="#">+ Add document</a> Aps0Xa781ToXcz6uiAHG ClYhe5Jp7g2ho2efiHzM FDekCZtS10xddM4NDXC1 QUESTIONS_LIST

QuizMaster Go to docs ?

## Cloud Firestore

[Data](#) [Rules](#) [Indexes](#) [Usage](#)

[Prototype and test end-to-end with the Local Emulator Suite, now with Firebase Authentication](#) [Get started](#)

[QUIZ](#) > [1pkgW3ucLe6s...](#) > 1 > [Aps0Xa781ToX...](#)

1pkgW3ucLe6skMC6lpLz	1	Aps0Xa781ToXcz6uiAHG
<a href="#">+ Start collection</a> 1 2 3 <a href="#">+ Add field</a> COUNTER: "4" NAME: "C++" SET1_ID: "1" SET2_ID: "2" SET3_ID: "3"	<a href="#">+ Add document</a> Aps0Xa781ToXcz6uiAHG ClYhe5Jp7g2ho2efiHzM FDekCZtS10xddM4NDXC1 QUESTIONS_LIST	<a href="#">+ Start collection</a> <a href="#">+ Add field</a> A: "making C++ operator work with objects" ANSWER: "4" B: "giving new meaning to existing operator" C: "making the new operator" D: "adding operation to the existing operators" QUESTION: "Operator overloading is ____ ?"



# **CONCLUSION**

## **7.1 The Application Achievements**

Quiz Master is a user-friendly, android based application that can educate as well as entertain users with a quick quiz on any category. The main objective of this app is it contains multiple categories to choose from, giving the client a benefit to learn various skills at a one stop location. The other best thing is that a countdown timer starts as the question is displayed and the user needs to answer the questions within a limited time frame to get points. At last, the final score is displayed to the user to assess their knowledge and provides another chance to improve their results.

## **7.2 What have we learnt?**

The application Quiz Master has been implemented successfully on different Android devices and this enhanced our confidence in Android development. This is the first time that we have developed an Android Application and this gave us an idea of an Android Application Development, it helped us learn its database firebase Firestore and we would like to learn more about its development in the coming future and would like to try and develop more apps in Android.

## REFERENCES

- <https://developer.android.com/docs>
- <https://docs.oracle.com/en/java/>
- <https://www.youtube.com/>
- <https://www.google.com/>
- <https://firebase.google.com/docs>
- <https://developer.android.com/guide/>

## APPENDIX

Source Code:

- Quiz Master App:

<https://github.com/shashank9046/Quiz-Master/tree/master/QuizMaster>

- Quiz Master Admin App:

<https://github.com/shashank9046/Quiz-Master/tree/master/QuizMasterAdmin>