

**A REPORT
ON
DOCUMENT MANAGEMENT SYSTEM ANDROID
APPLICATION**



BY

Deepak Jain

2017B5A30935P

AT



Podar Mills (NTCL), Mumbai
A Practice School-I Station of

BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI
(JUNE,2019)



**A REPORT
ON
DOCUMENT MANAGEMENT SYSTEM
ANDROID APPLICATION**

BY

Name	ID No.(s)	Disciplines
Deepak Jain	2017B5A30935P	MSc. Physics + B.E.(Hons.) Electrical and Electronics

Prepared in partial fulfillment of the
Practice School-I Course

AT

Podar Mills (NTCL), Mumbai

A Practice School-I Station of

**BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI
(JUNE,2019)**

**BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE
PILANI (RAJASTHAN)
Practice School Division**

Station: Podar Mills, NTCL

Centre: Mumbai

Duration: 2 months

Start Date: 21/05/2019

Date of Submission: 13/06/2019

Title: Document Management System Android System

Student:

Deepak Jain (2017B5A30935P), MSc. Physics + B.E.(*Hons*) Electrical and Electronics Engineering

Experts:

Mr P.T. Deshmukh (H.R. Manager)

PS Faculty:

Dr Abhishek Sarkar

Key Words:

Java Development, App Development, NoSQL querying, Database Design, OOP patterns, Firebase, UI/UX

Project Areas:

Android Development, Java, Android Studio, Firebase

Abstract:

An android app application for uploading, searching, and managing documents was proposed by us to be developed in the Android Studio using Java. This requires familiarity with Java development, OOP principles, Database Management and Cloud Storage.

Signature of Student:

Date :

Signature of PS Faculty:

Date:

Table Of Contents

1. Introduction	1
2. Technologies used	3
2.1. Android Studio.....	3
2.2. Firebase Realtime Database	4
2.3. Firebase Cloud Storage	5
2.4. Firebase Authentication.....	6
2.5. GitHub.....	7
3. Main Activities	8
3.1. Login Activity.....	9
4. Conclusion.....	10
5. Appendix	11
6. References... ..	12

1. Introduction

Information Technology has taken over all sectors, and we are constantly drifting towards the mobile era. The number of mobile phone users is greater than desktop users. While mobile devices like laptops, and tablets made it possible to work on the go, now workers can use their applications on their mobile devices to take advantage of business capabilities and insights.

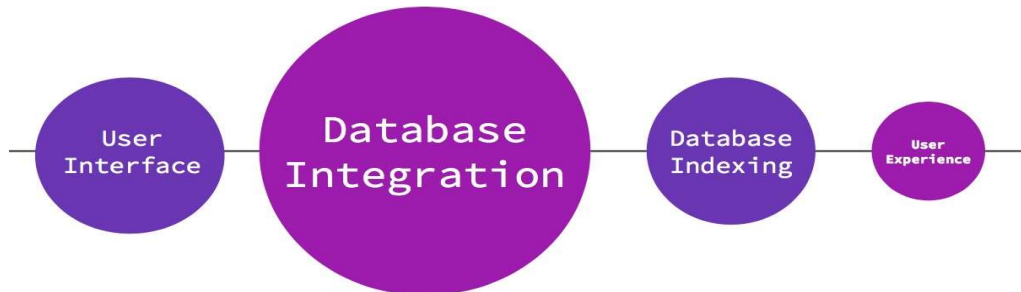
Our Project, Document Management System Android Application will offer its' users, i.e. the employees of Podar Mills various benefits. Currently, the system is manual. This restricts access to information and documents to the workplace only. Since the employees at Podar Mills have to quite often commute between offices and various other places the application will offer the information at the fingertips of the user. It will allow employees to improve quality of service to customers by giving them access to the relevant information quickly. Having information at the fingertips helps mobile workers deepen business relationships and response times. This application makes it easy for employees to enter all relevant data easily as they gather it in the field without computing the data into back end systems (potentially making errors along the way).

Currently, the system will be hosted through the Android application but later on can be expanded to iOS and web application as Firebase offers cross-platform connectivity. At a time, the application will support 100 simultaneous users and 5GB of document storage.

With this application, employees can take advantage of the functionality, data, and benefits of this application not only in the office, but from remote places also. Mobility is becoming more and more prevalent in the workspace, and hence it is important for the organizations to choose a system that can scale up their business and also have the flexibility and capability to integrate with new technology solutions.

Methodology:

Development Stage:



User Interface: Developing the basic User Interface to start with programming the logic. This includes making Bottom Navigation Bar, Buttons, Edit Texts and Dropdowns (Spinner) and linking up to the activities. We referred to various internet sources to help us design them. *Android Studio* has been used to code the User Interface. *Java* and *XML* have been used for UI designing.

Database Integration: Firebase Realtime Database is used for storing the User details, which includes the login credentials as well as document details. The documents are stored in Firebase Cloud Storage. The database is queried using *NoSQL* language.

Database Indexing: This technique helps in faster retrieval of data and can be used in sorting as well. Queries usually result in much better performance.

User Experience: Adding various features which will improve the user experience. This includes improving the User Interface as well as adding features like save passwords, uploading images by capturing through the mobile camera.

Post-Development: A beta version of the application will be released for the employees to find the various bugs present. After all these issues are resolved, the application will be deployed through the Google Play Store after completing documentation.

2. Technologies Required

2.1. Android Studio

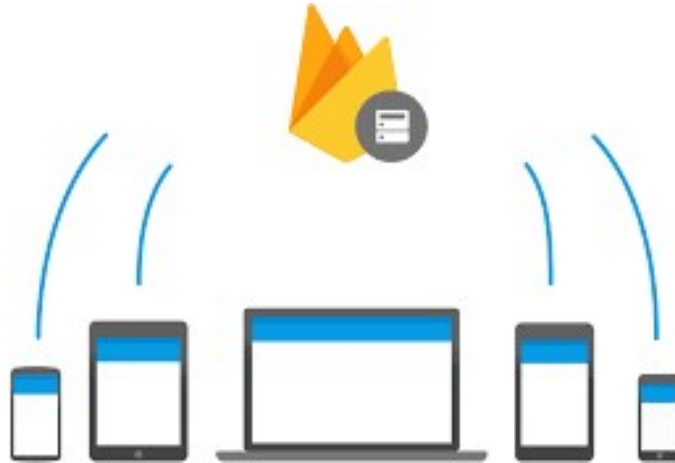


Android Studio is the official integrated development environment (IDE) for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. It is available for download on Windows, Linux and MacOS based operating systems. It is created as a replacement for the Eclipse Android Development Tools (ADT) as primary IDE for native Android application development. It has a powerful editor tool for developing creative UI/UX and emulators for different versions to test and simulate sensors without having actual Android devices. It also has a very intuitive Gradle plugin feature by which we can create application files (apks) with different configurations. Moreover, it makes exporting and uploading app bundles or apk on play store easy with a single click.

Languages used:

- *Java*
- *XML*
- *Kotlin*
- *NoSQL*
- *JSON*

2.2. Firebase Realtime Database



The **Firebase Realtime Database** is a cloud-hosted database. Data is stored as JSON and synchronized in realtime to every connected client. When you build cross-platform apps with our iOS, Android, and JavaScript SDKs, all of your clients share one Realtime Database instance and automatically receive updates with the newest data.

Instead of typical HTTP requests, the Firebase Realtime Database uses data synchronization—every time data changes, any connected device receives that update within milliseconds. Provide collaborative and immersive experiences without thinking about the networking code.

Firebase apps remain responsive even when offline because the Firebase Realtime Database SDK persists your data to disk. Once connectivity is reestablished, the client device receives any changes it missed, synchronizing it with the current server state. This is used to access and maintaining user information and setting up their power.

2.3. Firebase Cloud Storage



Cloud Storage for Firebase is a powerful, simple, and cost-effective object storage service built for Google scale. The Firebase SDKs for Cloud Storage add Google security to file uploads and downloads for your Firebase apps, regardless of network quality. Firebase Storage SDKs can be used to store images, audio, video, or other user-generated content.

Firebase SDKs for Cloud Storage perform uploads and downloads regardless of network quality. Uploads and downloads are robust, meaning they restart where they stopped, saving the users time and bandwidth.

Cloud Storage for Firebase is built for exabyte scale, so when the app goes viral, we can effortlessly grow from prototype to production using the same infrastructure that powers used in Spotify and Google Photos. This is used to store documents and to access them from anywhere

2.4. Firebase Authentication



Firebase Authentication provides a platform to authenticate users by providing backend services, SDKs, and ready-made libraries to authenticate users to the app. The method can vary from simple email/ password register to google, facebook, GitHub login using OAuth.

Implementing Firebase Authentication is a fast and easy than other methods of authentication. From the Firebase console, we just need to choose from the popular login methods that we want to offer in our app(like Facebook, Google, Twitter and email/password) and then add the Firebase SDK to our app. The app will then be able to connect securely with the real-time database, Firebase storage or to any other firebase service. If we have an auth system already, we can use Firebase Auth as a bridge to other Firebase features.

This feature of firebase is used in our app to identify the user and give particular access to a different user in the apps.

2.5. Github

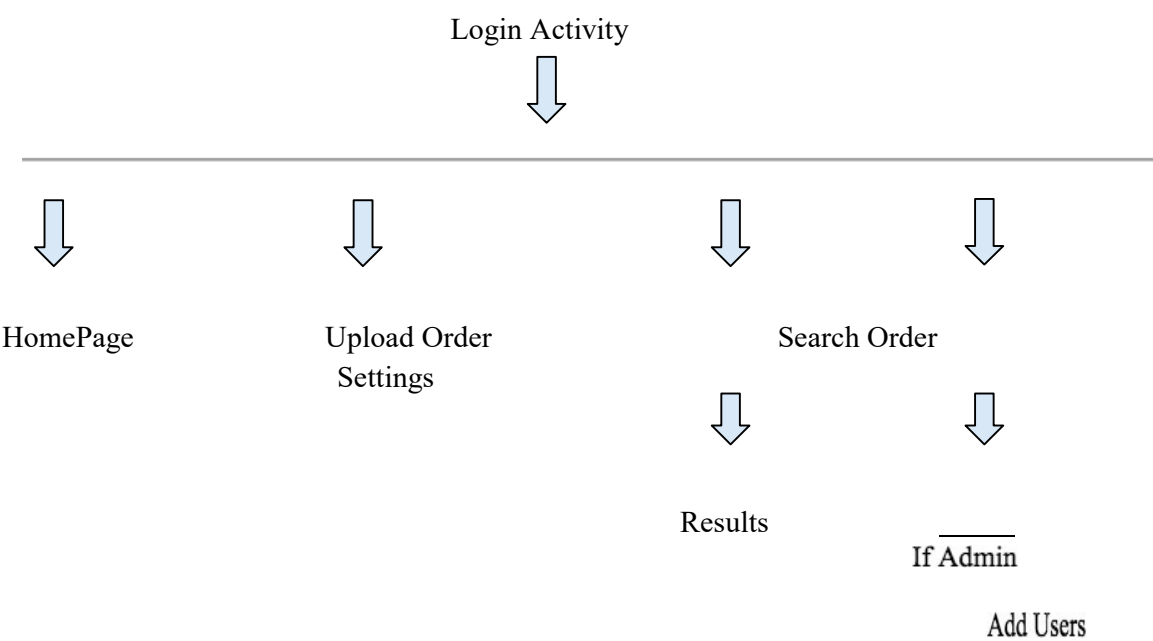


GitHub is a web-based repository hosting and managing service. It offers all of the distributed version access control and source code management functionality of Git as well as adding its features. Unlike Git, which is strictly a command-line tool, GitHub is basically a web GUI of git and desktop as well as mobile integration. It also provides access control and collaboration features such as bug tracking, issue reporting, pull requests, task management, and guidelines for every project.

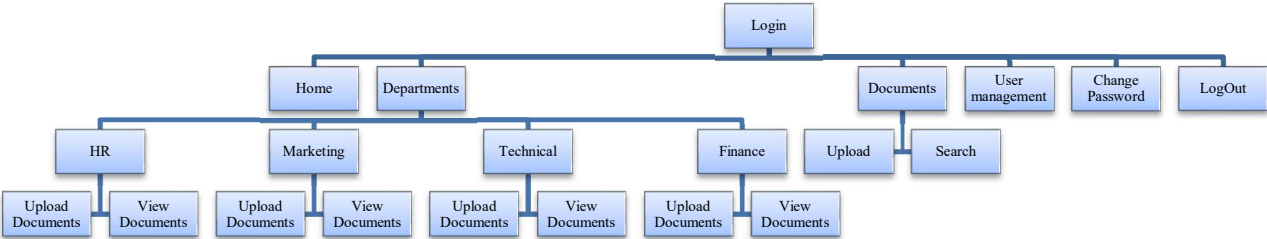
GitHub offers both plans, private repositories and free accounts, which are usually used to host open-source software projects. Projects on GitHub can be accessed and operated using the standard Git command-line interface and all of the standard Git commands such as push, pull, commit work with it. As of April 2018, GitHub reports having more than 16 million users and more than 55 million repositories, which makes it the largest community host of source code in the world. My git handle is djain454.

3. Main Activities

Flow Chart:



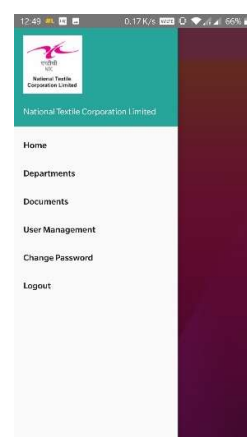
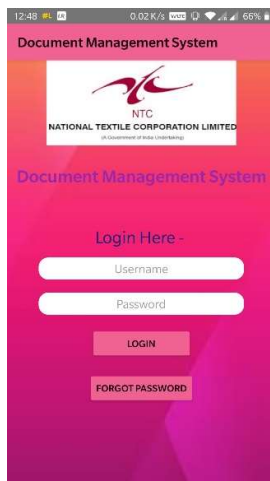
Structure:



3.1. Login Activity

This is the first page which will be displayed when the user opens the app. It has a basic layout with 2 EditTexts, 1 Button and 1 ImageView containing the NTC logo.

The 2 EditText contain the username and password which will be entered by the user, after that, there is a Login button which when clicked constructs a query to the Firebase realtime database and checks whether the entered username is even valid, or if valid is the entered password correct.



4. Conclusion

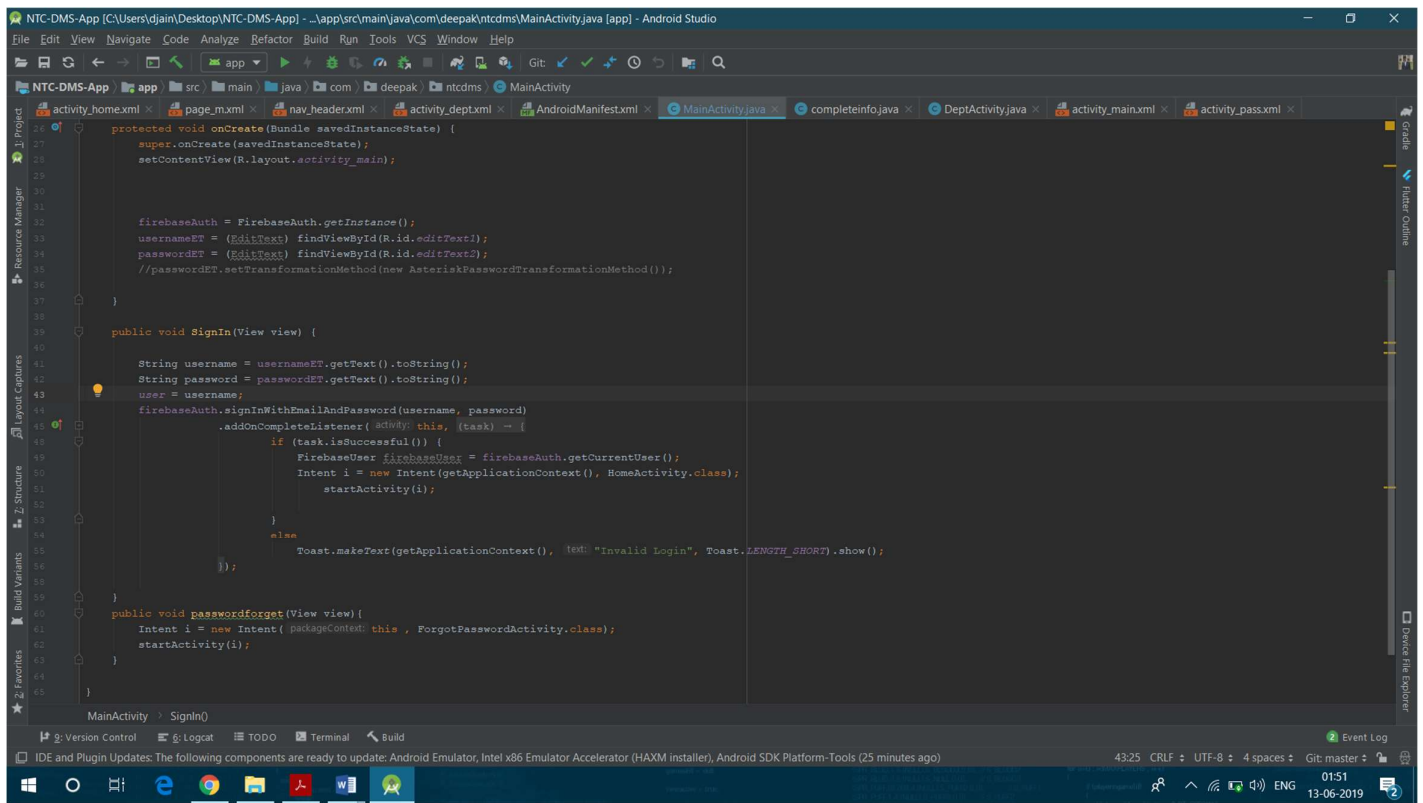
By making this document management system android application I will become well acquainted with software like Android Studio, Firebase Realtime Database, Firebase Storage and GitHub. The basic structure and my learning curve are ready to complete the project, and hence, the development speed will increase.

Here at the mill managing and keeping track of documents is a huge task, so this project will aid the company with uploading, searching or retrieving past documents and will make it very easy and uncomplicated. So, this app will help the company progress further and manage all their documents on the go.

5. Appendix

Code Snippets

Login:



```
18 protected void onCreate(Bundle savedInstanceState) {
19     super.onCreate(savedInstanceState);
20     setContentView(R.layout.activity_main);
21
22     firebaseAuth = FirebaseAuth.getInstance();
23     usernameET = (EditText) findViewById(R.id.editText1);
24     passwordET = (EditText) findViewById(R.id.editText2);
25     //passwordET.setTransformationMethod(new AsteriskPasswordTransformationMethod());
26
27 }
28
29 public void SignIn(View view) {
30
31     String username = usernameET.getText().toString();
32     String password = passwordET.getText().toString();
33     user = username;
34     firebaseAuth.signInWithEmailAndPassword(username, password)
35         .addOnCompleteListener(new OnCompleteListener<SignInResult>() {
36             if (task.isSuccessful()) {
37                 FirebaseAuthUser = firebaseAuth.getCurrentUser();
38                 Intent i = new Intent(getApplicationContext(), HomeActivity.class);
39                 startActivity(i);
40             }
41             else {
42                 Toast.makeText(getApplicationContext(), "Invalid Login", Toast.LENGTH_SHORT).show();
43             }
44         });
45
46 }
47
48 public void passwordforget(View view){
49     Intent i = new Intent( packageContext: this , ForgotPasswordActivity.class);
50     startActivity(i);
51 }
52
53 }
```

The screenshot shows the Android Studio interface with the MainActivity.java file open. The code implements a login function using Firebase Authentication. It includes a SignIn method that takes a View parameter, extracts username and password from EditText fields, and uses FirebaseAuth to sign in. If successful, it starts the HomeActivity; otherwise, it shows a toast message "Invalid Login". There is also a passwordforget method that starts the ForgotPasswordActivity. The IDE interface includes a toolbar at the top, a project structure pane on the left, and a bottom status bar showing IDE updates and system information.

6. References

1. Firebase Help Docs:_
<https://firebase.google.com/docs/>
2. Android Developers Help
Doc:_
<https://developer.android.com/>
3. Github Guides
<https://guides.github.com/>