Project Report

CONSULT DOCTOR

CSE3001-SOFTWARE ENGINEERING

Submitted by

17BCI0058 – SAURABH AMBAR

17BCE0451 – PARAG KAUNDIL

17BCE0107- ADITYA UNNAVA

Under the guidance of

Prof. Mythili T

Bachelor of Technology

in

Computer Science and Engineering

Spec. in Information Security

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

School of Computing Science and Engineering

APRIL,2019

TABLE OF CONTENTS

01. Title of Project

02. Members of Project

03. Motivation

04. Objective

05. Introduction

06. System Model

07. Modules Description

08. Implementation

09. Sample Report Of Acceptance Testing

10. Summary/Conclusion

11. Reference

01. CONSULT DOCTOR APP

03. Motivation

In India many people believe in physical meeting with doctor not the online consultation. But after the first meeting, visiting doctor for small queries becomes uneasy. Some people need to visit the doctor regularly until their condition gets better. This takes a lot of time from their day and schedule. Also, its difficult for people who live far from the hospitals to always visit a doctor.

Online Pathology Lab Report is a very useful for labs, hospitals and patients because it is much easier for labs to upload patient test reports online so that patient, doctor can view their lab reports without visiting hospital or laboratory. Whenever the lab report came, we need to visit to take that handy report. But once the report gets online updated, in the real time patient as well as doctor can view it.

Since all reports are uploaded on cloud in digital form hence it will reduce the use of paper on a large scale. Even it will reduce the risk of loss of handy documents. It is an attempt to bring innovative disruption in the diagnostic healthcare space by empowering patients and making diagnostic healthcare services convenient and accessible to them.

Our app is designed with all these concerns in mind.

04. Objective

- Resolves issues/queries of a patient via doctor consultation.
- Saves good time of patient.
- Patient can also take appointment from a doctor.
- 24x7 Availability

  Patients can access their lab reports anytime anywhere and on any smart device.

- Secured

  Each patient report is password protected which ensures high end to end security and takes care of patient privacy too.

- A Cost-Effective Solution

  Patient don’t need to visit the hospital for issues/queries and even for lab reports.

05. Introduction

This application platform aims to foster various health related issues/queries of a patient through consultation with a doctor. Understanding how best to use the application in a secure manner is something we want to make very easy and intuitive. What's more, knowing how to talk to your doctor regularly can be difficult so here is a way out for how to communicate with. For ensuring any kind of misinterpretations, doctor/dietician consultation will be available at their leisure periods.

Some people need to visit the doctor regularly until their condition gets better. This consume a lot of time of their day and schedule for appointment. Also, it’s difficult for people who live far from the hospitals and often need to visit a doctor until their condition gets better. So, this is how our app works in order to resolve all the above-mentioned issues.

In this project, we are working on making an app which will help the patients to communicate with the doctors. After the first examine of patient, he will get a system generated user id and password through which it can contact the doctors for any queries related to the illness.

This app is useful for doctors to have in regular touch with their patients and also help him in scheduling next appointment date if required.

This app will save the time of both patient and doctors.

This app will have additional Features of informing about new emerging and some common diseases with their preventive methods and some home remedies.

If the patient has the lab test then the lab report is shown in the patient and doctors' portal.

Here we go…

06. System Model

a) Process Model: **Spiral Model**

- Proposed by Barry Boehm in 1988
- It is a combination of prototype and sequential or waterfall model.
- It is used for generating the software projects. This model is a risk driven process model.
- Every phase in the Spiral model is start with a design goal and ends with the client review.
- The development team in this model begins with a small set of requirements and for the set of requirements team goes through each development phase.
- The development team adds the functionality in every spiral till the application is ready

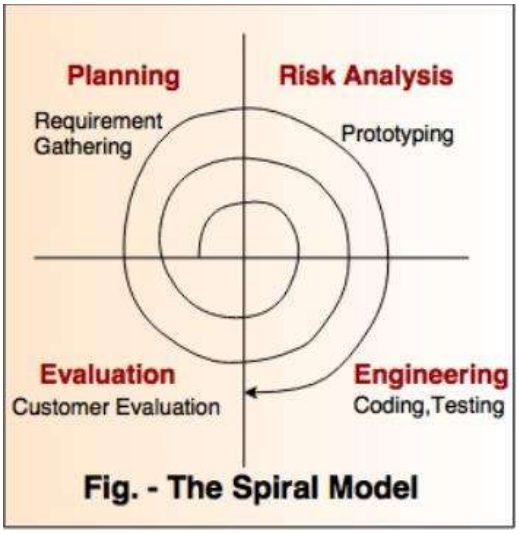


Fig. - The Spiral Model

Four Phases of Spiral Model:

1) Planning

This phase, studies and collects the requirements for continuous communication between the customer and system analyst.

It involves estimating the cost and resources for the iteration.

2) Risk Analysis

This phase, identifies the risk and provides the alternate solutions if the risk is found.

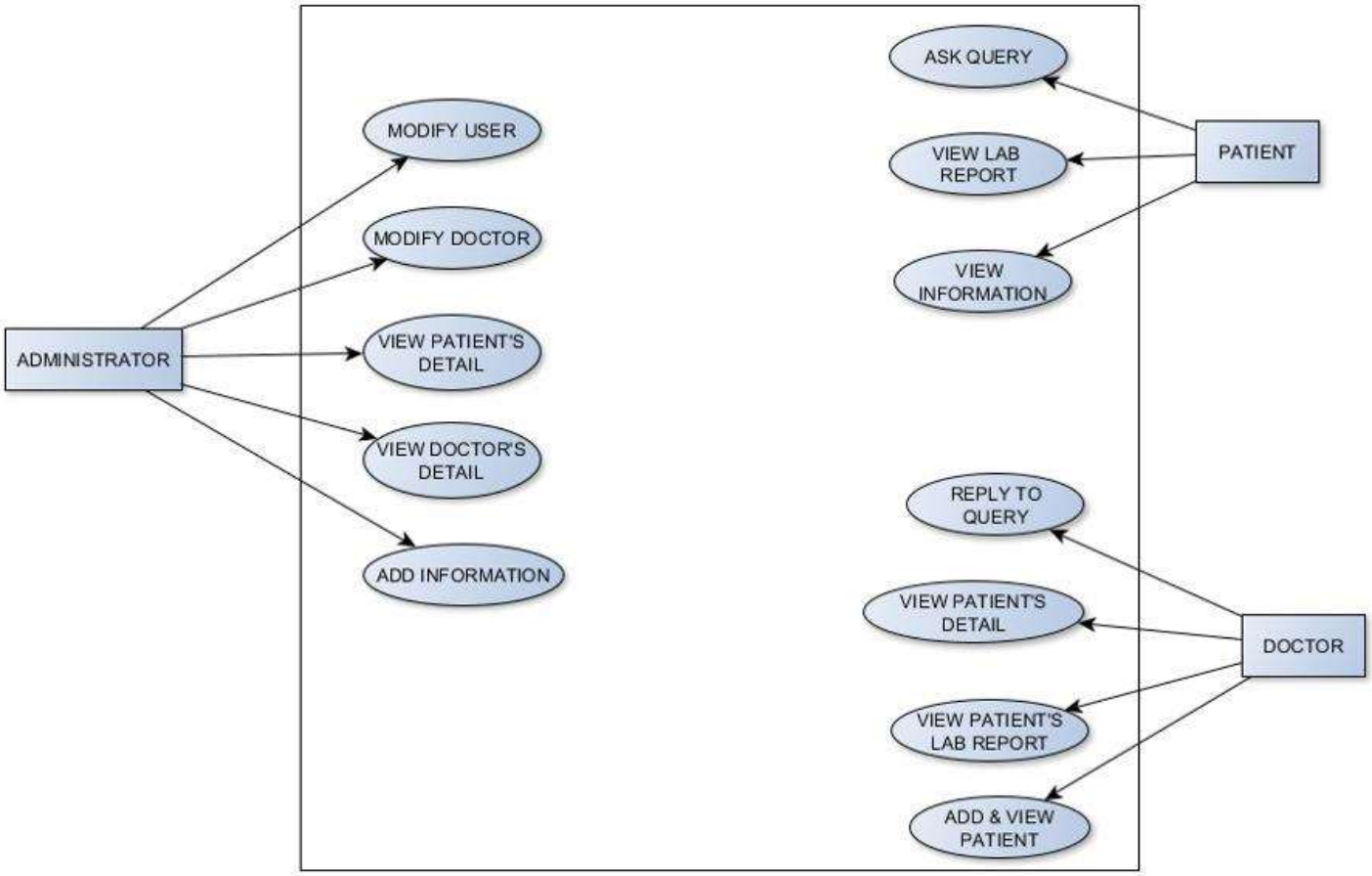
3) Engineering

In this phase, actual development i.e coding of the software is completed. Testing is completed at the end of the phase.

4) Evaluation

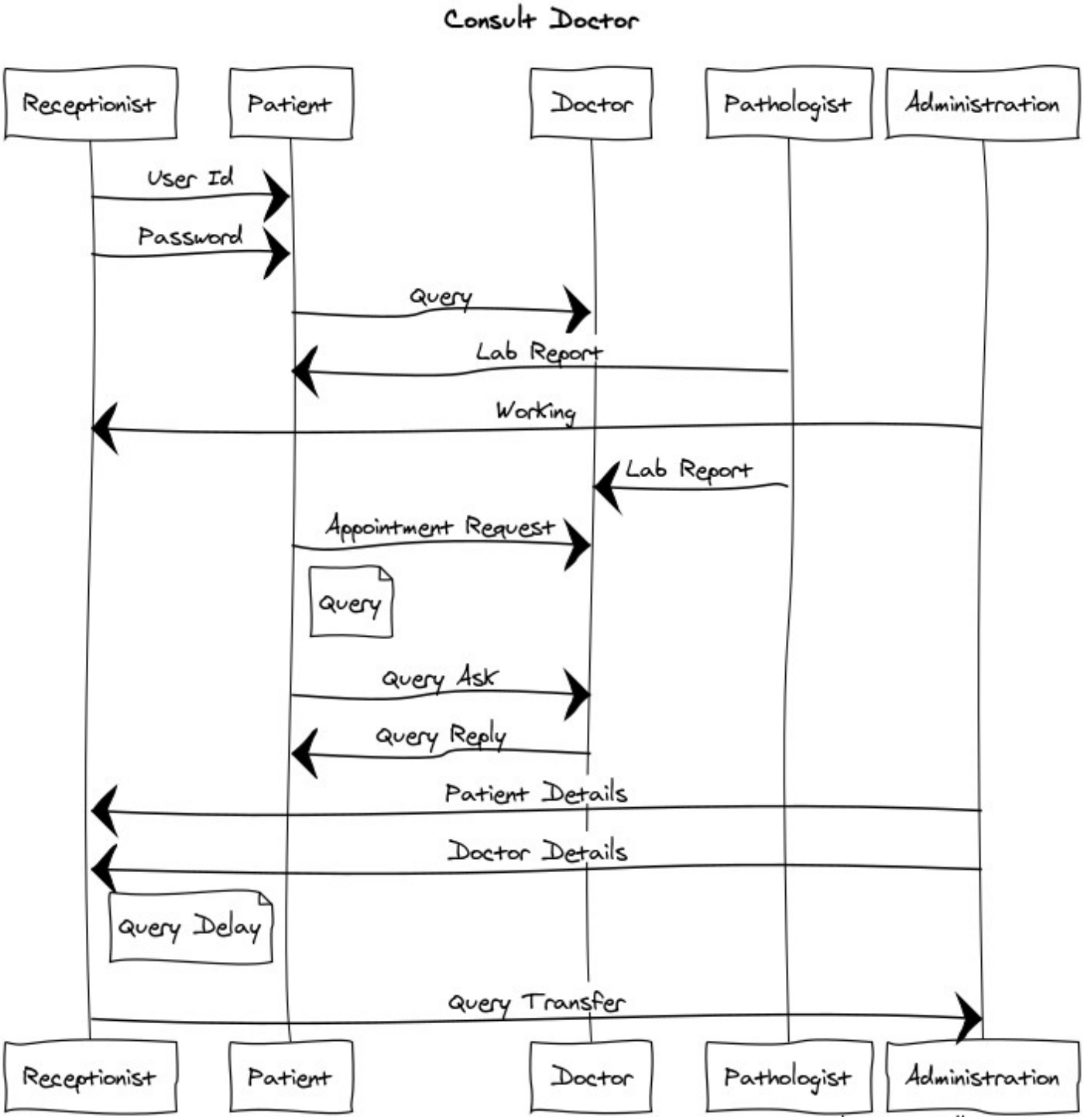
Get the software evaluated by the customers. They provide the feedback before the project continues to the next spiral.

b) Use Case Model:



USE CASE MODEL

c) Sequence Diagram :



d) Behavioral viewpoint: interactive model

- The objective of behavior modeling is to study/specify the behavior of the objects in a system.

- An OO system is a system made up of objects that work together to realize some desirable functionality.
  – We have the desired functionality -> use cases
  – We have the object structure -> classes

- At the logical level, the behavior models allow us to:

– Complete the structural model by finding the methods of our classes.

– Validate the structural model by making sure that all required attributes and (navigable) associations are present

- At the physical level,

– Sequence & Activity diagrams define a specification for our algorithms.

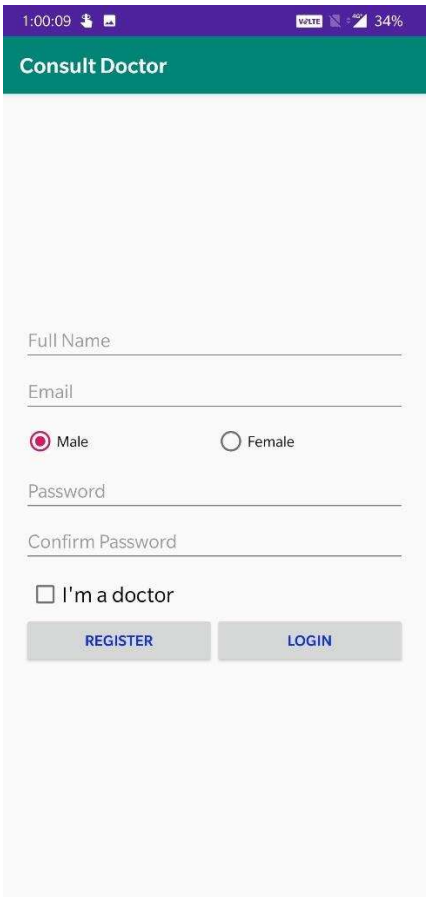
– State Machines can be used to generate executable code

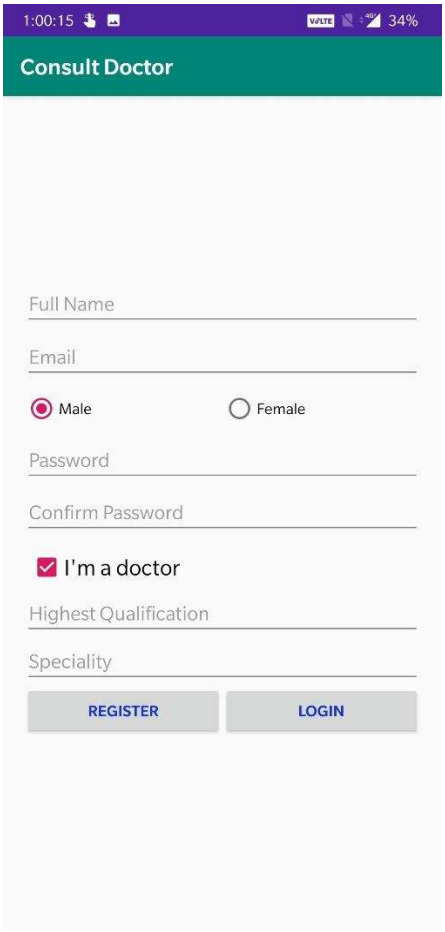
07. Modules Description

a) Authentication Page: Login, Register

This is the common login portal for patient and doctor.

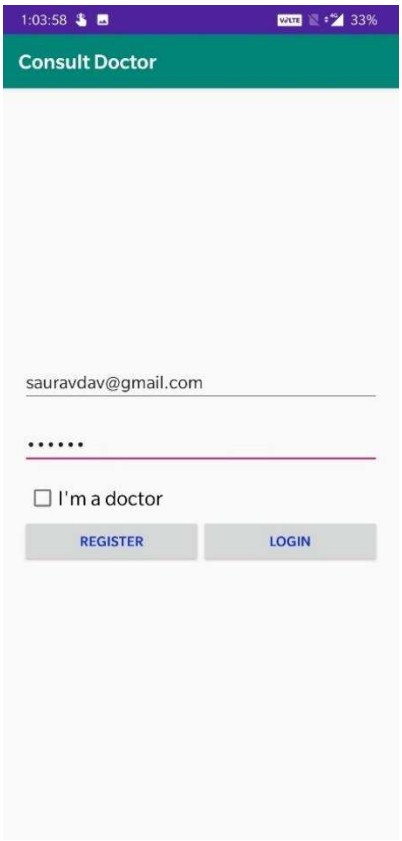
This is the register page for patient.





This is the register page for doctor. Here doctor need to provide Qualification and Speciality.

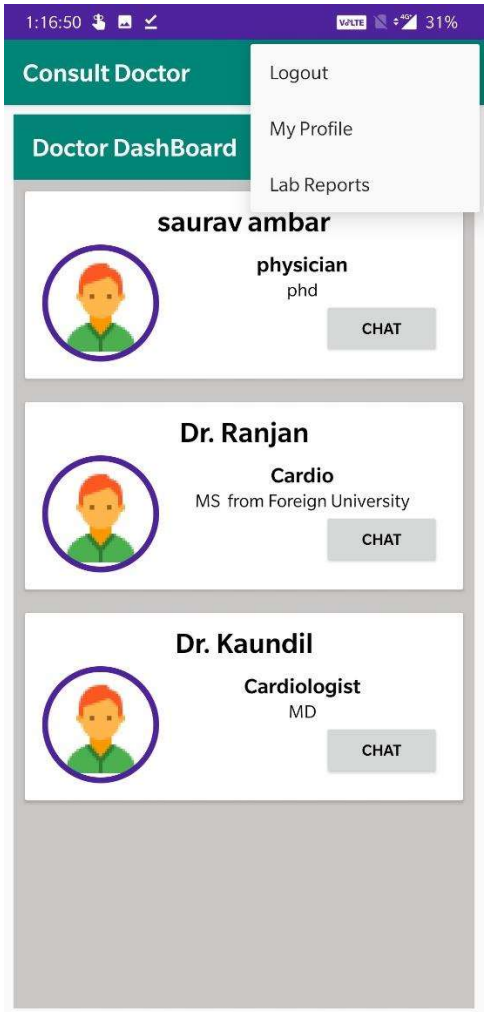
b) Patient Module: Doctor List, Patient Messaging List

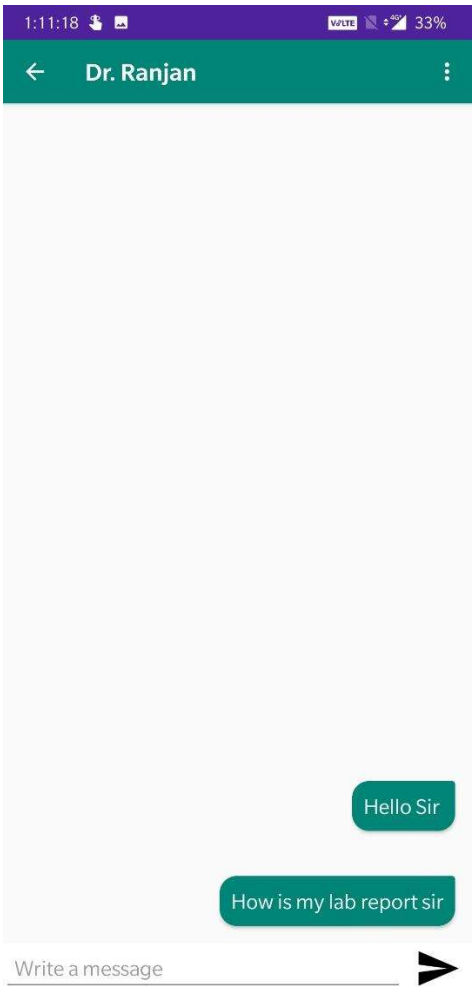


Login using patient credential.

List of doctors shown in patient portal.

At the corner there is option for viewing the Lab Report.



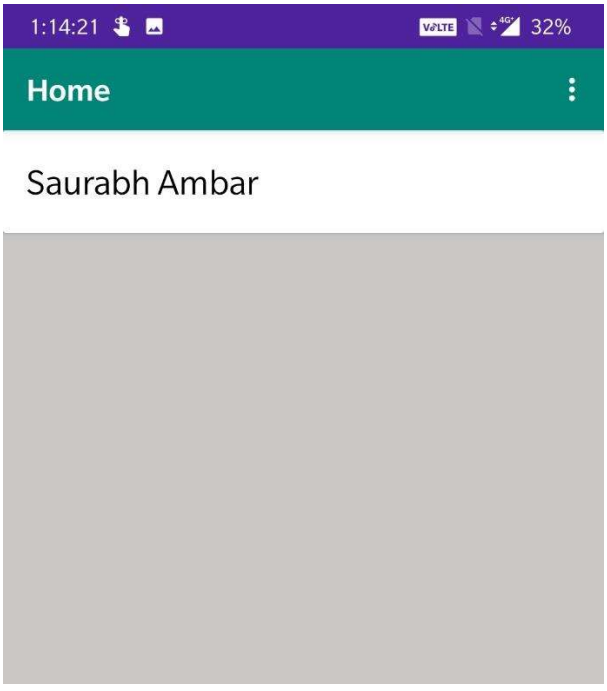


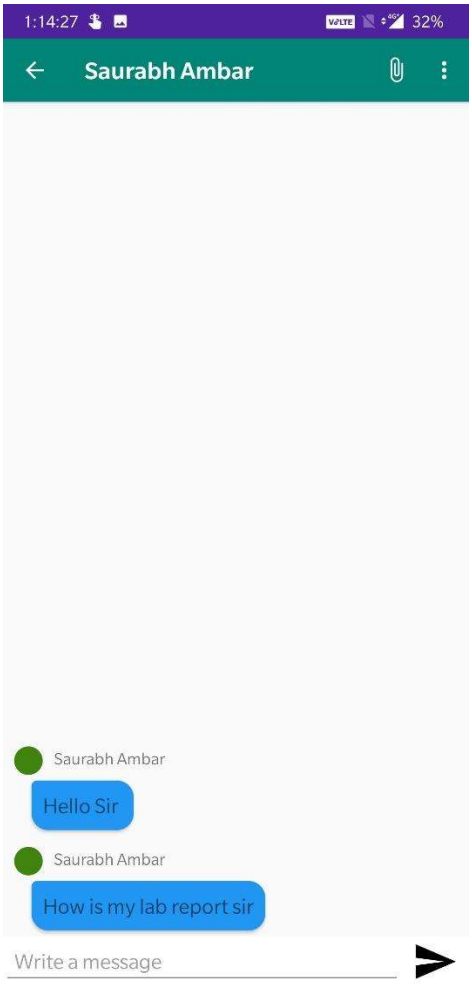
Message sent to Dr. Ranjan

c) Doctor Module: Doctor Messaging List

Login using doctor credential.

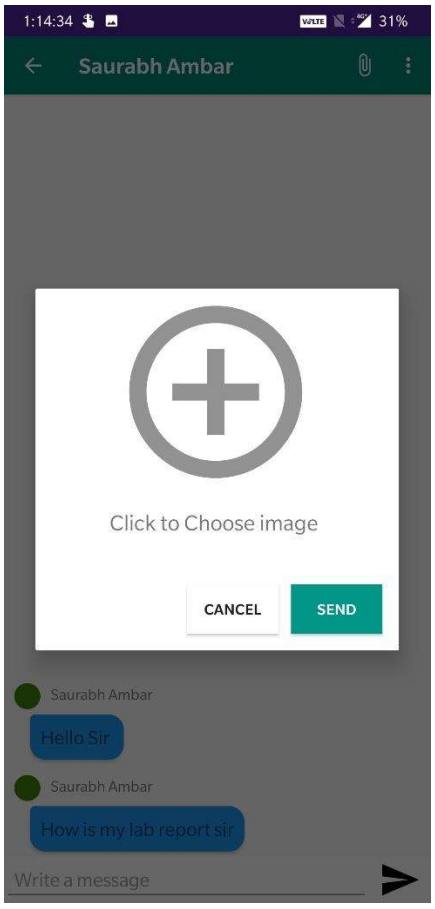
Message from the patient.



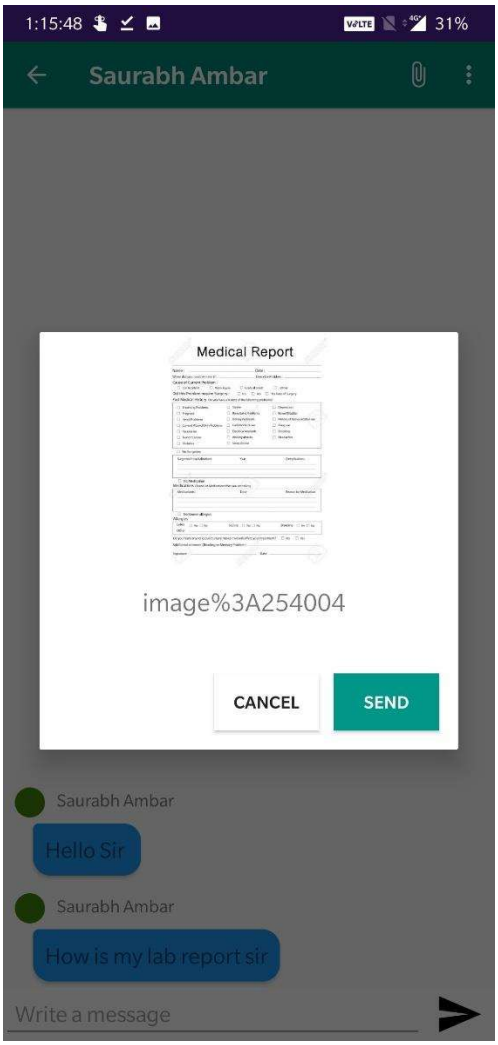


Message box of the patient, from here doctor can reply to patient queries.

Uploading Lab Reports.

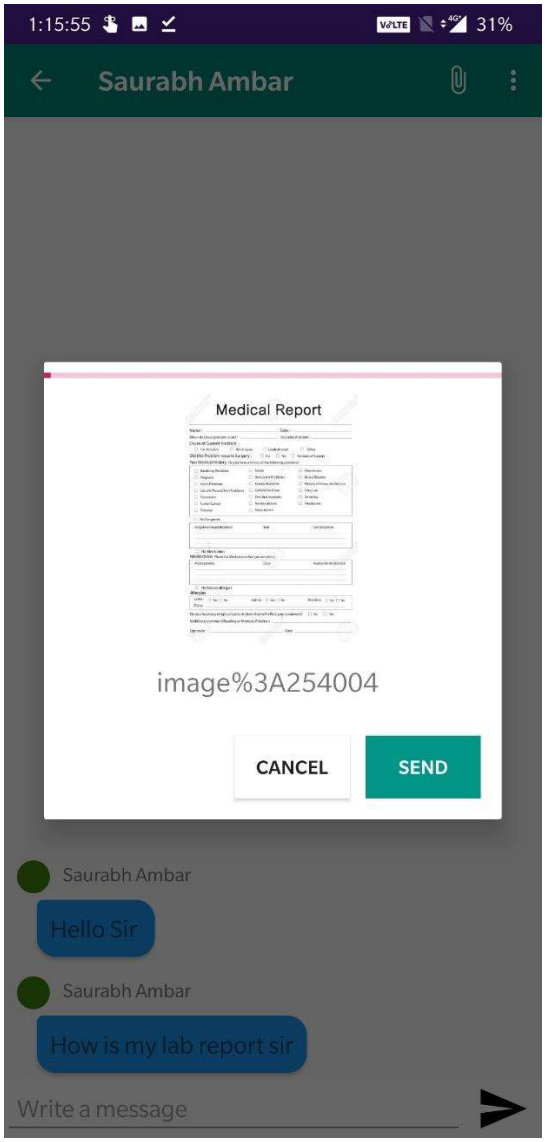


a) Lab Report Upload Module : Lab Report Patient

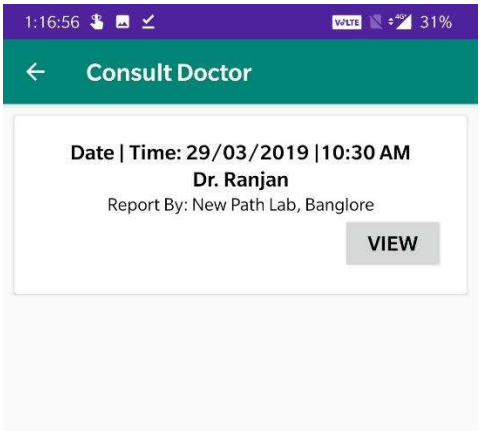


Uploading the report.

Red mark shows the uploading of the documents.



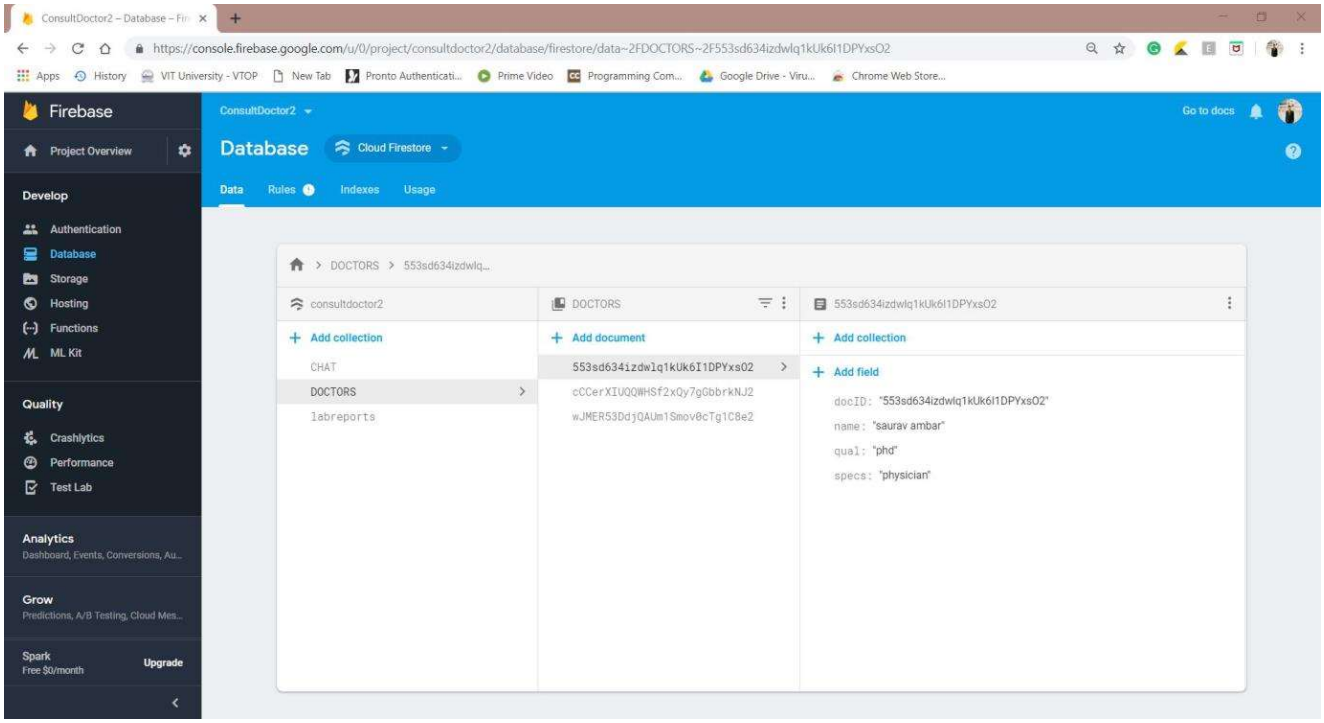
Conformation of Lab Report sent to Patient.

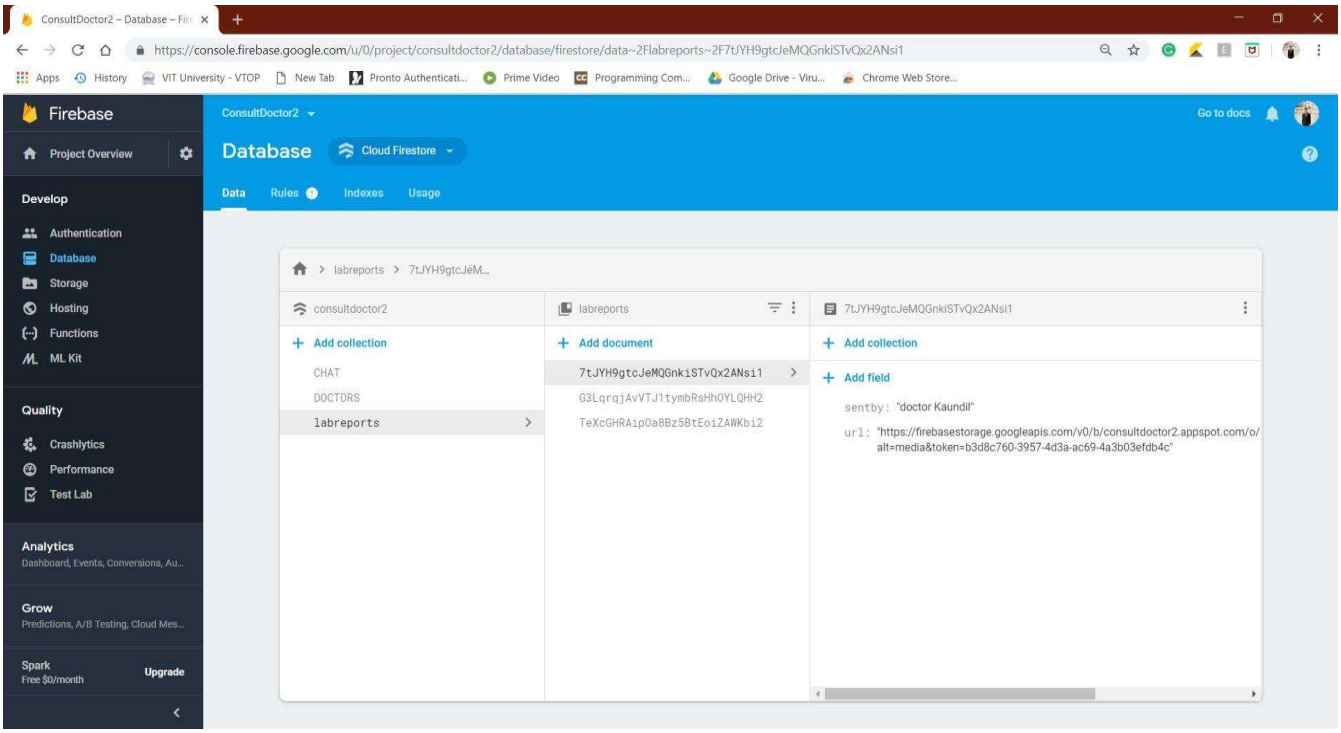


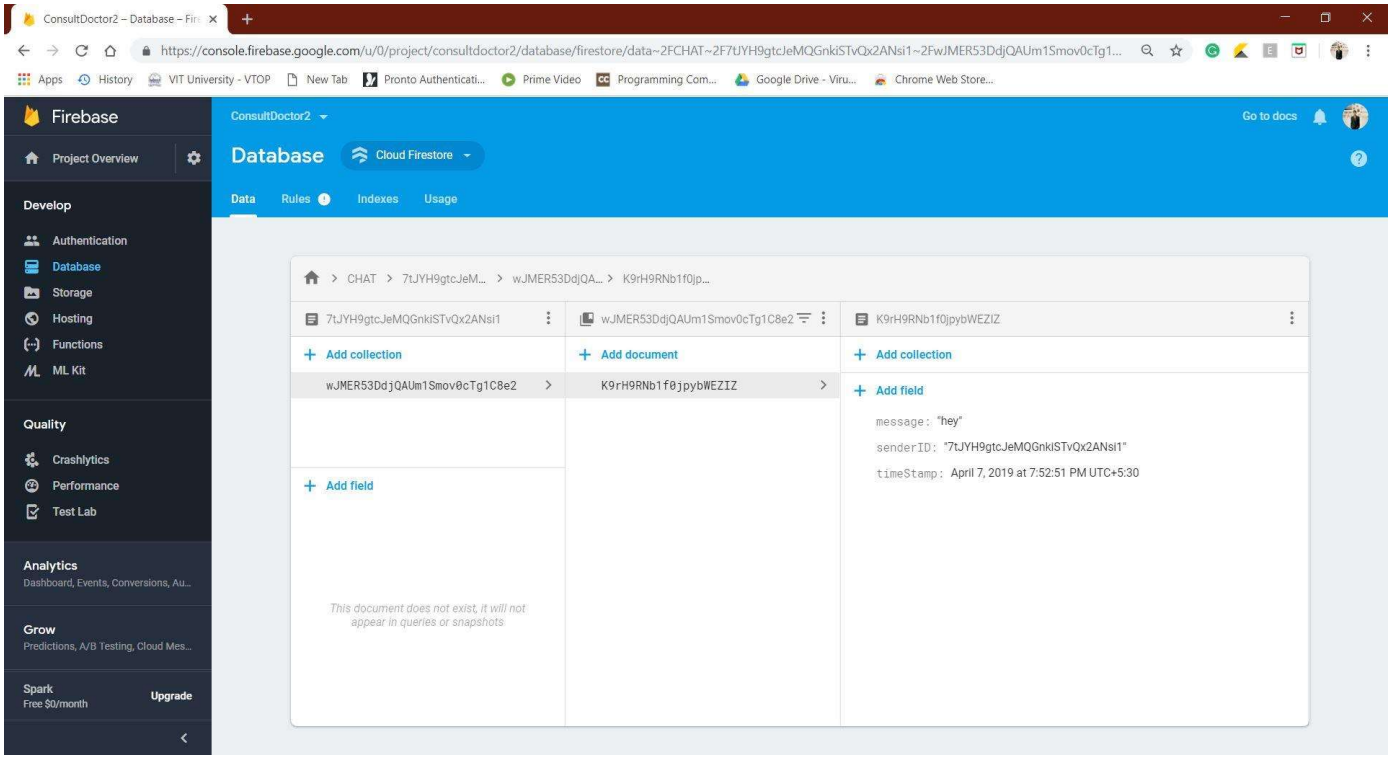
Patient Portal view for lab report

View of Lab Report from Patient Portal.

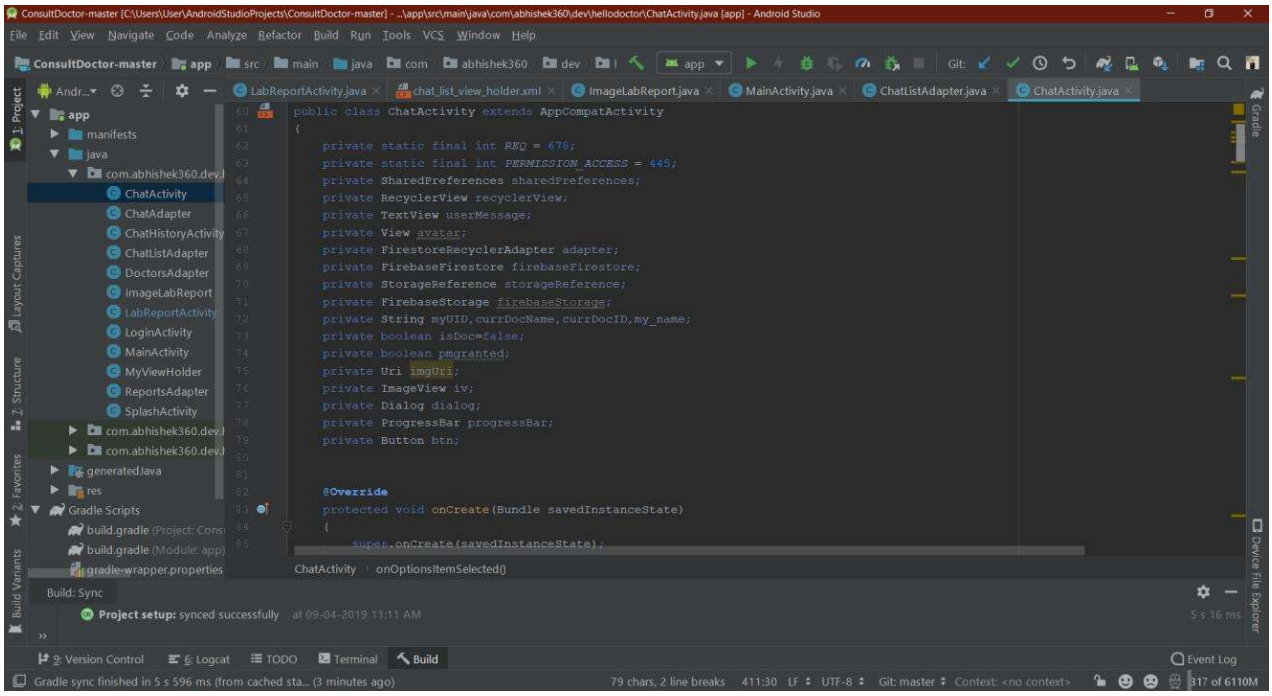
b)Firebase Establishment

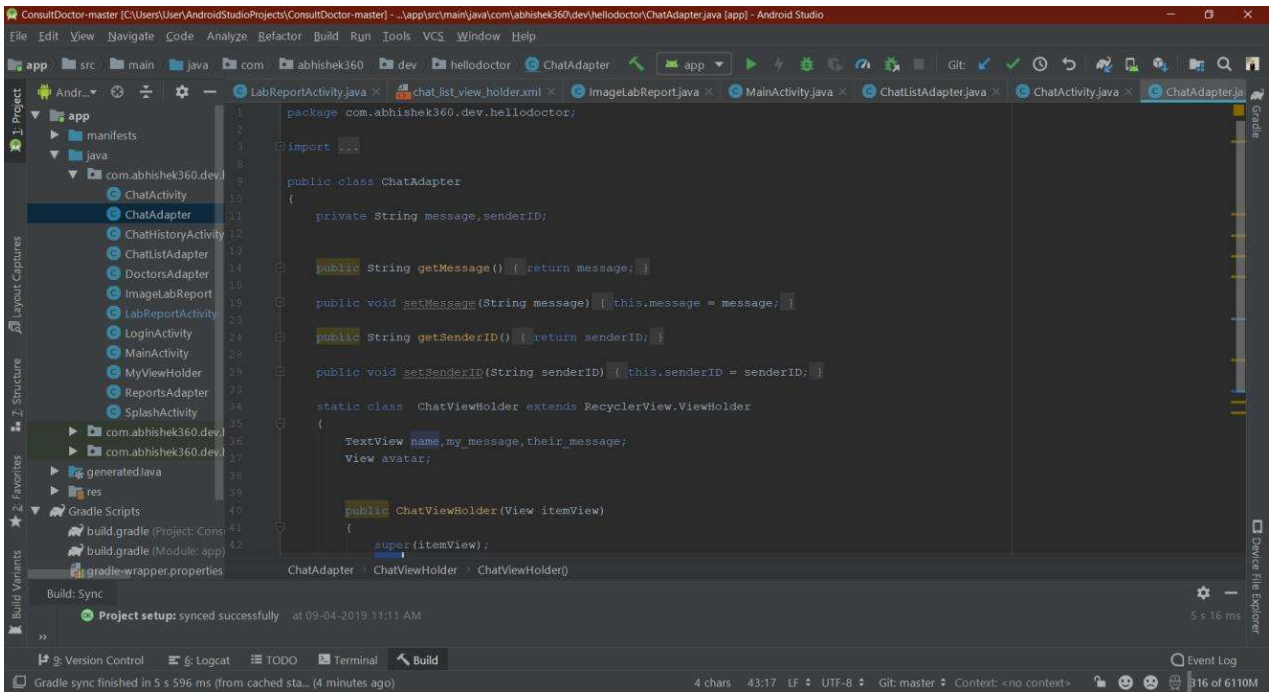


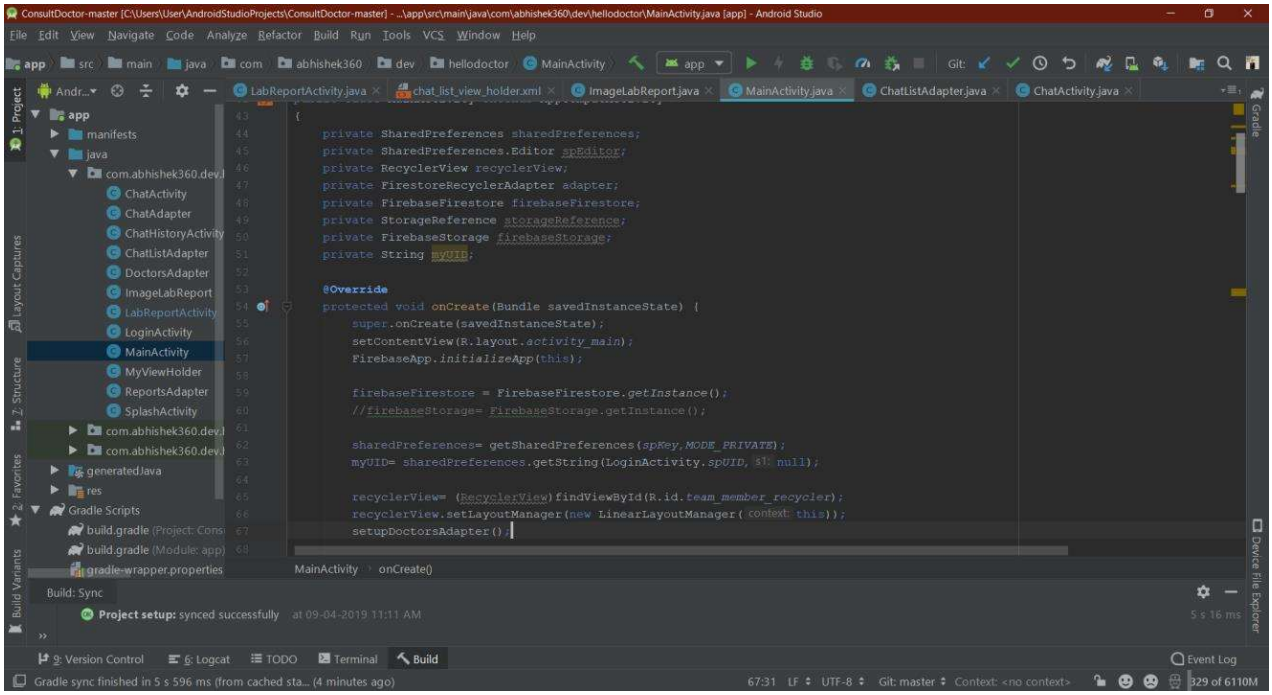


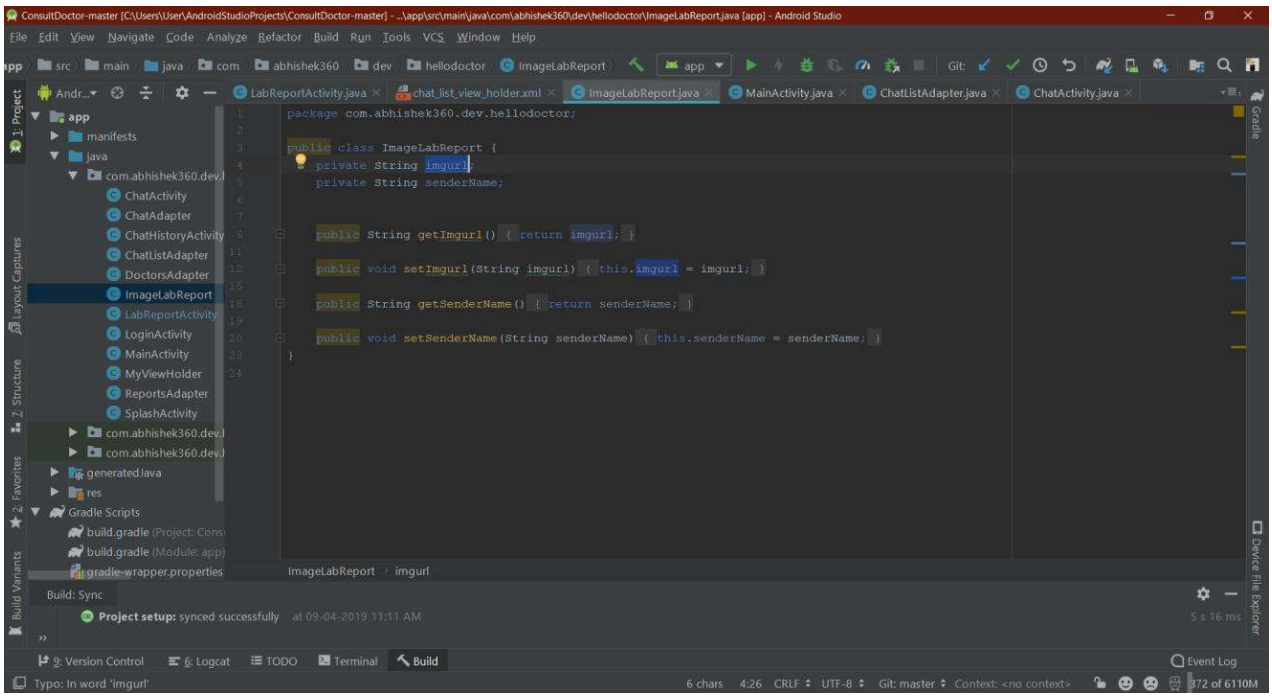

ConsultDoctor2 – Database – Fire
https://console.firebase.google.com/u/0/project/consultdoctor2/database/firestore/data~2FCHAT~2F7tJYH9gtcJeMQGnkiSTvQx2ANsi1~2FwJMER53DdjQAUm1Smov0cTg1...
Apps
History
VIT University - VTOP
New Tab
Pronto Authenticati...
Prime Video
Programming Com...
Google Drive - Viru...
Chrome Web Store...
Firebase
Project Overview
Develop
Authentication
Database
Storage
Hosting
Functions
ML Kit
Quality
Crashlytics
Performance
Test Lab
Analytics
Dashboard, Events, Conversions, Au...
Grow
Predictions, A/B Testing, Cloud Mes...
Spark
Free $0/month
Upgrade
ConsultDoctor2
Go to docs
Database
Cloud Firestore
Data
Rules
Indexes
Usage
CHAT > 7tJYH9gtcJeM... > wJMER53DdjQA... > K9rH9RNb1f0jp...
7tJYH9gtcJeMQGnkiSTvQx2ANsi1
wJMER53DdjQAUm1Smov0cTg1C8e2
K9rH9RNb1f0jpybWEZIZ
Add collection
Add document
Add collection
wJMER53DdjQAUm1Smov0cTg1C8e2
K9rH9RNb1f0jpybWEZIZ
Add field
Add field
message: "hey"
senderID: "7tJYH9gtcJeMQGnkiSTvQx2ANsi1"
timeStamp: April 7, 2019 at 7:52:51 PM UTC+5:30
This document does not exist, it will not appear in queries or snapshots

08. Implementation






ConsultDoctor-master [C:\Users\User\AndroidStudioProjects\ConsultDoctor-master] - ...\app\src\main\java\com\abhishek360\dev\hellodoctor\MainActivity.java [app] - Android Studio
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
LabReportActivity.java
chat_list_view_holder.xml
ImageLabReport.java
MainActivity.java
ChatListAdapter.java
ChatActivity.java
private SharedPreferences sharedPreferences;
private SharedPreferences.Editor spEditor;
private RecyclerView recyclerView;
private FirestoreRecyclerAdapter adapter;
private FirebaseFirestore firebaseFirestore;
private StorageReference storageReference;
private FirebaseStorage firebaseStorage;
private String myUID;
@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);
FirebaseApp.initializeApp(this);
firebaseFirestore = FirebaseFirestore.getInstance();
//firebaseStorage= FirebaseStorage.getInstance();
sharedPreferences= getSharedPreferences(spKey,MODE_PRIVATE);
myUID= sharedPreferences.getString(LoginActivity.spUID, s1: null);
recyclerView= (RecyclerView)findViewById(R.id.team_member_recycler);
recyclerView.setLayoutManager(new LinearLayoutManager(context: this));
setupDoctorsAdapter();
MainActivity › onCreate()
Build: Sync
Project setup: synced successfully at 09-04-2019 11:11 AM
9: Version Control 6: Logcat TODO Terminal Build Event Log
Gradle sync finished in 5 s 596 ms (from cached sta... (4 minutes ago)
67:31 LF UTF-8 Git: master Context: <no context>
329 of 6110M


ConsultDoctor-master [C:\Users\User\AndroidStudioProjects\ConsultDoctor-master] - ...\app\src\main\java\com\abhishek360\dev\hellodoctor\ImageLabReport.java [app] - Android Studio
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
LabReportActivity.java
chat_list_view_holder.xml
ImageLabReport.java
MainActivity.java
ChatListAdapter.java
ChatActivity.java
package com.abhishek360.dev.hellodoctor;
public class ImageLabReport {
private String imgurl;
private String senderName;
public String getImgurl() { return imgurl; }
public void setImgurl(String imgurl) { this.imgurl = imgurl; }
public String getSenderName() { return senderName; }
public void setSenderName(String senderName) { this.senderName = senderName; }
}
ImageLabReport › imgurl
Build: Sync
Project setup: synced successfully at 09-04-2019 11:11 AM
9: Version Control 6: Logcat TODO Terminal Build Event Log
Typo: In word 'imgurl'
6 chars 4:26 CRLF UTF-8 Git: master Context: <no context>
372 of 6110M

09. Sample Report of Acceptance Testing

Test the program and find any bugs or glitches in the chat application. Testing will be done as the application is being developed, but the more developed it gets, the more testing will be done.

UNIT TESTING

Unit testing is a software development process in which the smallest testable part of an application, called units, are individually and independently scrutinized for the proper operation. There are three modules in the system that we are working upon. On elf them is then login model where the user authentication takes place and the next on the registration module. The third module is the major module that we are going to use in this system that is messaging conversation module. This module is the place where the data in converted into the encrypted form and save into database.

The test case for the module are written and the results are noted down. These tests and the results can be used to check if the system is working according to needs and requirements that are provided.

Unit Id/ Name	User authentication

Test Case Id	Test Case Description	Expected Result	Actual Result	Remarks
1.	Valid Email and Password	User Login in	PASS	
2.	Invalid Email & Password	Reload Page	PASS	
3.	Valid Email and in-valid Password	Reload Page	PASS	
4.	Valid Password and Invalid email	Reload Page	PASS	

Table 1

User Id/Name	User Registration

Test Case Id	Test Case Description	Expected Result	Actual Result	Remarks
1.	Valid Email and Username and Password and Re-past Password	Show Login Page and Success Message	PASS	
2.	Repeated Email	Error Message	PASS	
3.	Repeated Username	Error Message	PASS	
4.	Invalid Database Connectivity	Error Message	PASS	

Table 2

Assumptions and Dependencies

• We can assume that the user can be able to use the system for the entire situation used. The user has full knowledge of the capabilities of the system. The user has to go through the user manual in order to use the full features of the system. The user has to know the constraints and has full knowledge on how to use the system completely.

• The cache on the browser is free when the software is being used.

10. Summary/Conclusion

The main objective of the project is to develop a Secure Chat Application between Patient, Doctor and Lab Assistant. I had done range of survey in order to achieve all the tasks, where I came to know about some of the products that are existing in the market. I have done a detailed research in that path to cover the loop holes that existing systems are facing and to eradicate them in our application. In the process of research, I came to know about the latest technologies and different algorithms.

Future Scope

With the knowledge I have gained by developing this application, I am confident that in the future I can make the application more effectively by adding this service.

1. Extending this application by providing Authorisation service.

2.Creating Database and maintaining users.

3.Adding end to end encryption to chat feature.

4.Adding push notifications.

11. Reference

- https://www.tutorialride.com/software-testing/iterative-spiral-model.htm
- https://www.tutorialride.com/software-testing/software-development-process-models.htm
- firebase.google.com/docs
- developer.android.com