**SMART GOVERNANCE**

**A Project Report**

**Submitted by:**

**RIDDHI PATEL (2014033800084133)**

**VRUNDA PATEL (2014033800084195)**

**MALAY JIVANI (2014033800084052)**

in partial fulfillment for the award of the degree

of

**BACHELOR OF ENGINEERING**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**

at

****

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**FACULTY OF TECHNOLOGY & ENGINEERING**

**THE MAHARAJA SAYAJIRAO UNIVERSITY OF BARODA**

**MAY 2018**

**DECLARATION**

We hereby declare that the project entitled “Smart Governance” submitted for the B. E. (CSE) degree is my original work and the project has not formed the basis for the award of any other degree, diploma, fellowship or any other similar titles.

**Signature of the Student**

**Place:**

**Date:**

**CERTIFICATE**

This is to certify that the project titled “Smart Governance” is the bona fide work carried out by Riddhi Patel,Vrunda Patel and Malay Jivani, students of B.E. (CSE), Department of Computer Science & Engineering, Faculty of Technology & Engineering, The Maharaja Sayajirao University of Baroda of during the academic year 2017-18, in partial fulfillment of the requirements for the award of the degree of Bachelor of Engineering (Computer Science and Engineering ) and that the project has not formed the basis for the award previously of any other degree, diploma, fellowship or any other similar title.

**Signature of the Guide**

**Signature of Head of the Department**

**Place:**

**Date:**

TABLE OF CONTENTS



|  |  |  |  |
| --- | --- | --- | --- |
| 1. | ABSTRACT | | 5 |
| 2. | GOALS AND OBJECTIVES | | 5 |
| 3. | ABOUT THE SYSTEM | | 6 |
| 4. | SYSTEM DEVELOPMENT STRATEGIES | | 8 |
| 5. | FEASIBILITY STUDY | | 9 |
| 6. | MODULES | | 10 |
|  | * 1. Visitor | |  |
|  | 6.2 User | |  |
|  | 6.3 Admin | |  |
|  | 6.4 Worker | |  |
| 7. | PROJECT PROFILE | | 12 |
|  |  | 7.1 System Requirements |  |
|  |  | 7.1.1 For developers |  |
|  |  | 7.1.2 Device Requirements |  |
|  |  | 7.2 Functional Requirements |  |
|  |  | 7.3 Non Functional Requirements |  |
| 8. | APPLICATION TECHNOLOGIES | | 13 |
| 9. | ANALYSIS AND DESIGN | | 18 |
|  | 9.1 Use Case Diagram | |  |
|  | 9.2 E R Diagram | |  |
| 10. | HOW TO USE THE APPLICATION( with screen shots) | | 21 |
| 11. | DATA DICTIONARY | | 41 |
| 12. | APPLICATION ANALYSIS | | 45 |
| 13. | FUTURE ENHANCEMENTS | | 47 |
| 14. | BIBLIOGRAPHY | | 47 |
|  |  | |  |



1. ABSTRACT:



With the constant increase in the technology and internet it is just a matter of time before we will be able to find everything easily. Imagine a situation where you are going to the government organization for a specific work. But employees are not helping to pass your file without bribe. Now you are helpless and don’t know to whom you should complain. It provides an online solution for filing complaints from anywhere. The user can register to the site to issue a complaint. Also, he can see the status of his complain. Searching for contact information of the different departments of the Municipal office is also possible and this does not require logging in to the system.

2. GOALS AND OBJECTIVES:



To make the hectic work of filing complains and keeping a track of it easy. Smart governance can be put on the App Store as a free application, because there are no third-party tools or plugins used in the application. With this project we aim at developing a convenient and generic complaining portal to enable the citizens to file complains in a hassle free manner.

The purpose of admin side is managing and assigning the complains to the workers of the respective departments. Analyze the work done by the worker and accordingly assign work.

3. ABOUT THE SYSTEM:





The system is a mobile based application. The system is made using Android Studio. The back end is handled by FireBase. In our application the visitor can visit our application for getting details about Municipality offices, they can also see the contact details of Municipality offices for their use, they can also contact us for knowing about our services and also can complain about any issue regarding the application but they can’t issue their complain as a visitor on Smart Governance application. If they want to complain he/she must be register himself/herself as a user and get a user-id for their next use of the application or further complain, and if they are already registered then they can directly issue their complain. We also give them an option to select the department. If they want details of any Municipality office (department) than they can easily get the information.

In our application, the user doesn’t have to stand in queue to register their complain, and don’t have to waste their time by going every day to government office if they want to know status of their complain. They just have to go through the application .They just have to register themselves on the application and select an option from menu “Complain” and in complain section, they have to select type of complain and have to fill up complain form. In complain form if user wants to share any extra information like images about complain, they can.

After user register’s the complain successfully then the complain goes to head of the complain department and the head of department assign the complain to workers of department by Email, call or personally. If the workers have smart phones they can use the application and get notification for the work assigned by their head. So then they can work on complain. After assignment of complain to worker, the admin can track the worker. So worker cannot avoid the complain or cannot waste their work hours in any activity which is not related to complain or their work. And main advantage of our application is user can see their complain status.

We are also giving the Data Analysis facility to the Municipality officers so they can take decision regarding development of area, workload, work distribution, worker’s efficiency for development of city. We also provide the official announcements and circular so the user of the application gets notified if their area or city have any announcements/circular. So the civic person will not face any difficulty regarding to government actions like some road construction work in progress, water supply cut off for some hours, electric supply cut off, any important person arrives our city.

This system is very easy to use and operate for all common users, governments officers and high authorities of Municipality, which is mainly important for the people who don’t have any proper information of the process of complain in municipality offices. So here we are implementing a municipality application with the best and more good facility for user and officers.

4. SYSTEM DEVELOPMENT STRATEGIES (SDLC):



There are mainly three types of system strategies. They are:

1. System Development Strategy
2. Structured Analysis Development
3. System Prototype

Out of the three strategies we’ve chosen the SDLC approach. It mainly consists of two major steps:

* System Analysis
* System Design

SDLC consists of the following activities:

1. Preliminary Investigation
2. Determination of System Requirements
3. Design of the System
4. Development of Software
5. System Testing
6. Implementation & Evaluation

5. FEASIBILITY STUDY:



An important outcome of the preliminary investigation phase of the SDLC is the determination that the system required is feasible. There are three aspects in feasibility study portion of investigation.

The main aim of feasibility study that we carried out was to explore whether the application is technically, economically and operationally feasible or not. Keeping these

three aspects in mind, we carefully evaluated our project to be feasible and well organized. This study leads us to following results:

TECHNICAL FEASIBILITY

The application needed to be implemented in as many android versions as possible. As most of the android development API come with backward compatibility we can target most popular latest versions of Android.

OPERATIONAL FEASIBILITY

The application user interface must be easy to all users. Users must be able to specify tasks and required location with minimum input requirements.

ECONOMIC FEASIBILITY

The application must be freely and easily available to users. This can be done by putting the application on App Stores. There is no development cost as android being open source technology.

6. MODULES:



The users are being classified according to their role in the system. In this system there are two types of users, and four main modules which are as follows:

6.1Visitor:

* Can see contact details and information of Municipality offices.
* Can contact us.
* Can see the news feed.
* To complain they must register.

6.2 User:

* No need to stand in a queue.
* Can easily complain their issues related to government.
* Can see complain status.
* Can get information about important announcements of city.
* Can reopen complain if they are not satisfied with service of municipality.
* Can contact us or municipality offices.

6.3Admin:

* Can see all complaints.
* Can see report analysis.
* Can assign work to workers.
* Can see complain status.
* Can create analysis report.
* Maintain the time limit for work .
* Update the status of the complains.
* Search the complaints using unique complain id .
* Ask the worker for the pending work by sending message to his account.

6.4Worker:

* Can see the complaints assigned to him.
* He receives mail from the admin about the issues regarding his department.
* He can search for the complaints from the complains assigned to him by using unique complain id.
* Can update the remarks.
* Can inform the admin about the progress of the work .
* Can reach the location where there is issue.

7. PROJECT PROFILE:



7.1. SYSTEM REQUIREMENTS

* + 1. FOR DEVELOPERS

|  |  |
| --- | --- |
| Android Version | Version :minSDKVersion 15  TargetSDKVersion 26 |
| Android Studio | **Windows OS**   * Microsoft Windows 7/8/10 (32-bit or 64-bit) * 2 GB RAM minimum, 8 GB RAM recommended * 2 GB of available disk space minimum, 4 GB Recommended (500 MB for IDE + 1.5 GB for Android SDK and emulator system image) * 1280 x 800 minimum screen resolution * JDK 8 |
| Internet Connectivity | Server needs connection to send emails and use cloud storage. |
| Back-End Database | Firebase console |

* + 1. DEVICE REQUIREMENTS (RECOMMENDED)

|  |  |
| --- | --- |
| Android Version | Min Version: minSDKVersion 15  TargetSDKVersion 26 |
| Internet Connectivity | Smart phone requires to have full internet connectivity ,so the application can connect to the server. |

* 1. FUNCTIONAL REQUIREMENTS
* File complains
* Check status
* Assign work
* Update status
* Search complains
  1. NON-FUNCTIONAL REQUIREMENTS
* Authentication
* Security
* Portability
* User friendly

8. APPLICATION TECHNOLOGIES:



ANDROID STUDIO

Android Studio is Google's officially supported IDE for developing Android apps. Android Studio is freely available under Apache License 2.0. The most recent stable version, 3.0.1, includes the following features:

* A unified environment where you can develop for all Android devices.
* Support for building Android TV apps and Android Wear apps.
* Template-based wizards to create common Android designs and components.
* A rich layout editor that lets users drag-and-drop user interface components, and that offers an option to preview layouts on multiple screen configurations.
* Android-specific refactoring and quick fixes.
* [Gradle](http://www.javaworld.com/article/2093605/build-ci-sdlc/evolving-gradle-build-from-ant-build-importing-ant-build-files.html)-based build support.
* Lint tools to catch performance, usability, version compatibility, and other problems.
* ProGuard integration and app-signing capabilities.
* A fast and feature-rich emulator.
* Instant Run to push changes to your running app without building a new APK.
* Built-in support for Google Cloud Platform, enabling integration with Google Cloud Messaging and App Engine.
* C++ and NDK support.
* Plugin architecture for extending Android Studio via plugins.

FIREBASE CONSOLE

Firebase is a development framework that supports web applications, iOS applications, and, unsurprisingly, Android applications. That means developers not only have access to a broad spectrum of tools to make their lives easier, they also have the ability to use Firebase to produce applications that run on multiple operating systems.

**Analytics**

The biggest difference between Google Analytics and Firebase Analytics is the metrics that are tracked. Instead of page views, impressions, and session lengths, developers will see how many users downloaded the app and their behavior within it. That way, designers can improve their apps to enhance the user experience and, of course, increase conversions.Firebase Analytics provides developers with access to more than 500 metrics and enables audience segmentation based on devices, events, and other user properties.

**Cloud Messaging**

FCM is a cross-platform messaging solution that enables developers to deliver messages to client apps. Even better: it’s offered by Google at no cost.

**Authentication**

Firebase Authentication delivers a suite of tools that make it easy for developers to authenticate and authorize users. It also offers single sign-on support with identity providers like Facebook, Twitter, and, obviously, Google. The Firebase authentication solution also integrates with well-known security standards.

**Database**

It’s a cloud-based NoSQL database solution. Data is stored in JavaScript Object Notation (JSON) format for easy retrieval without the complicated hassles of inner and outer joins associated with SQL queries.

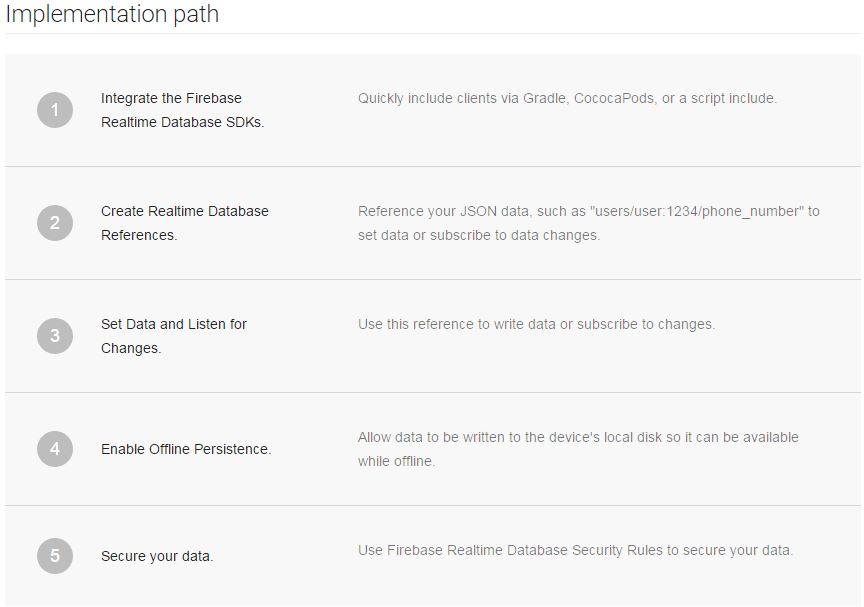
The Firebase Realtime Database also performs up-to-the-milisecond synchronization. That means when data changes, all devices running the app immediately receive the update. It’s a solution that gives users in different geographic regions the opportunity to communicate, collaborate, and coordinate more effectively.

**Crash Reporting**

Developers will not only receive notifications about when their apps crash, they’ll also receive full reports that offer actionable tips to help them diagnose and resolve software defects.

As of this writing, though, Crash Reporting is in beta release. That’s because there some known issues with Android and iOS platforms.

When it’s finally delivered, Crash Reporting will give developers the ability to monitor fatal and non-fatal errors, collect relevant data needed to diagnose the problem that caused the crash, and integrate crash stats with Firebase Analytics..



Also, when a device goes offline, all of its Firebase apps will sync with the database once the device is online again. That process happens behind the scenes in a way that’s completely invisible to the user.

**Storage**

If Firebase app users need to store videos, photos, or other types of large binary files.Even better: uploads and downloads are robust. That means if they’re interrupted at any point for any reason, they’ll resume where they left off later on.

**Hosting**

Firebase Hosting makes it easy to answer that question for business owners who’ve already made the decision to use Firebase as a development platform. That’s because it’s a production-grade content-delivery system that’s offered as part of the Firebase suite.The Firebase hosting solution offers security, fast content delivery, and the ability to rapidly deploy an application to production. It also offers one-click rollbacks.

**Remote Configuration**

Firebase Remote Config enables you to change the design and behavior of your app without publishing an update.It works like this: developers begin by creating in-app default values that control the look and feel of the app as well as its behavior. Then, they can later use Firebase console to override those defaults and create a richer user experience.

**Testing**

The testing environment also gives developers the opportunity to test on real devices instead of just emulators. That’s helpful because there’s a myriad of different types of Android devices and configurations currently on the market.

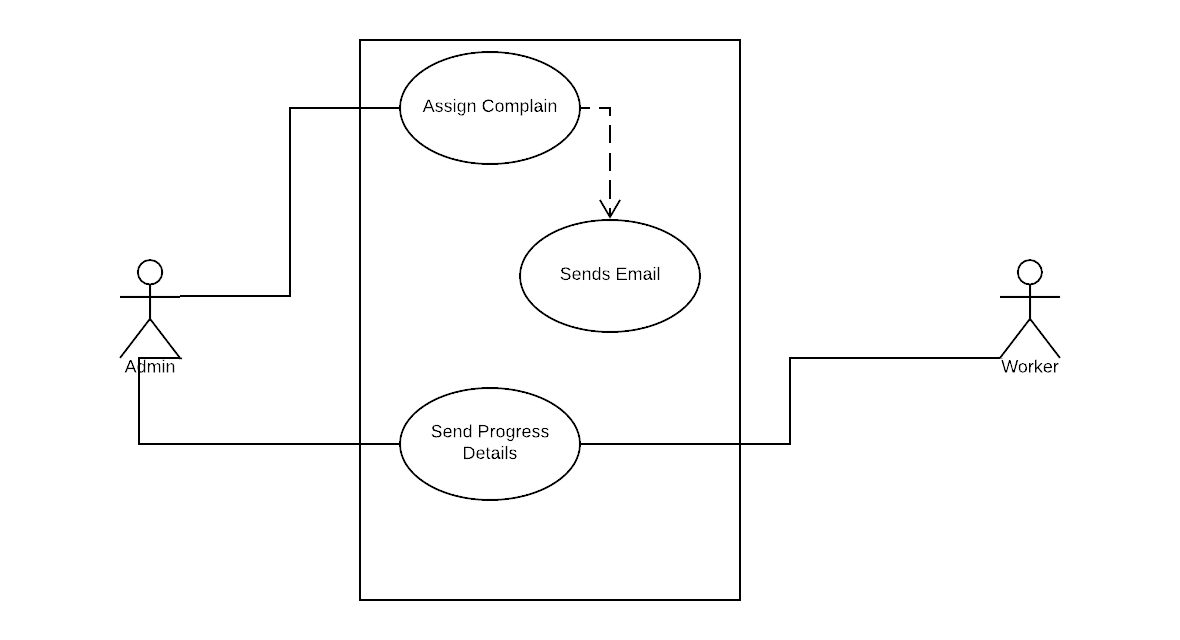
Also, developers will have the opportunity to use Robo test to run tests on apps even if they haven’t written any testing code. Robo test analyzes the code itself and performs tests relevant to the user interface.

9. ANALYSIS AND DESIGN:

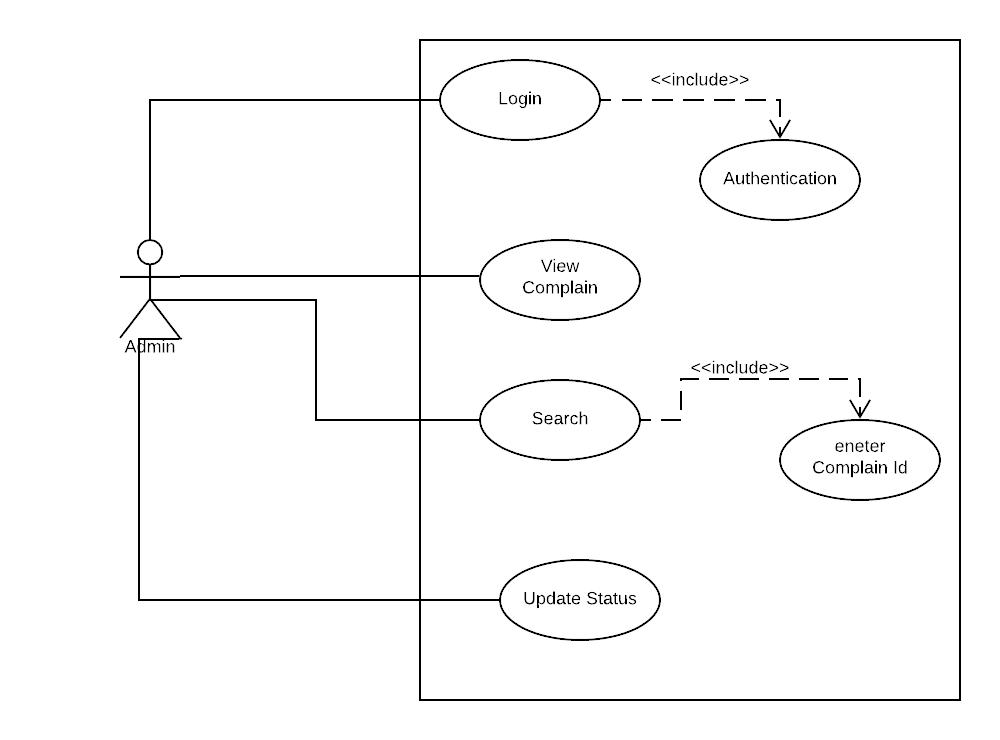


9.1 USE CASE DIAGRAM

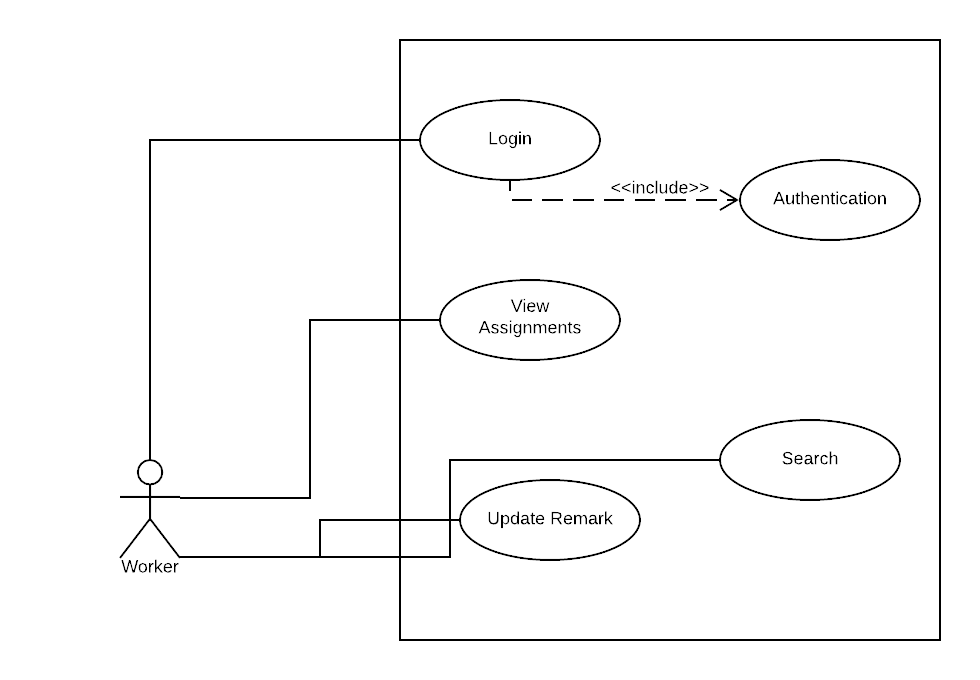
Use case for communication between worker and admin.



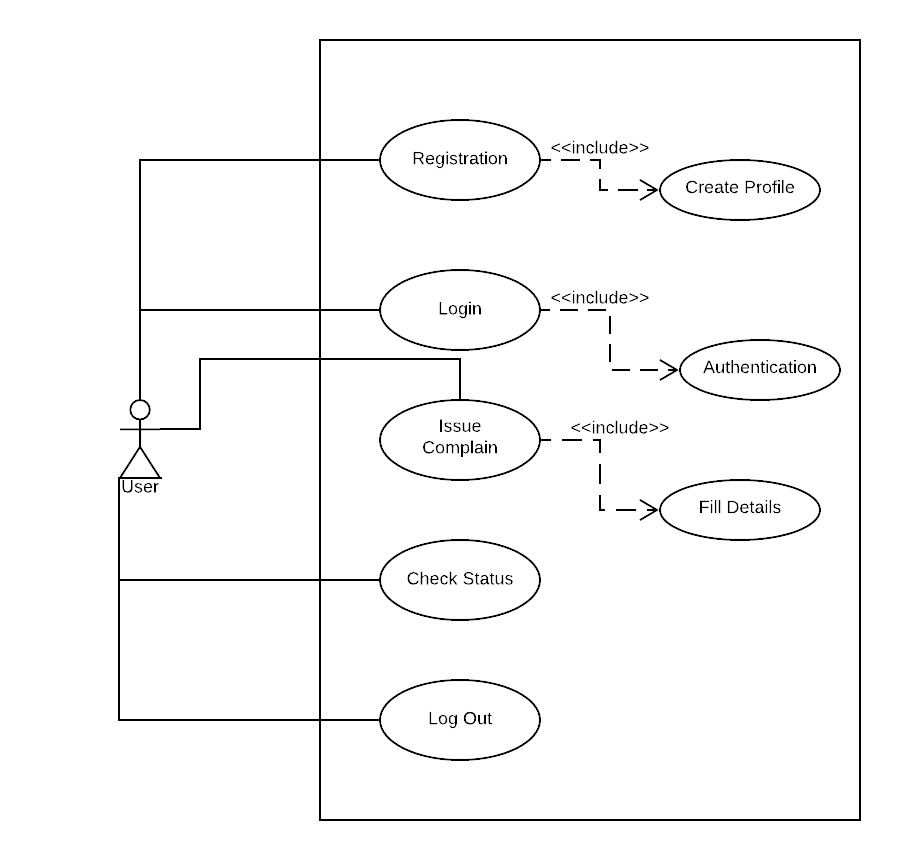
Use Case for admin activities



Use Case for worker activities



Use Case for user activities



9.2 E-R DIAGRAM

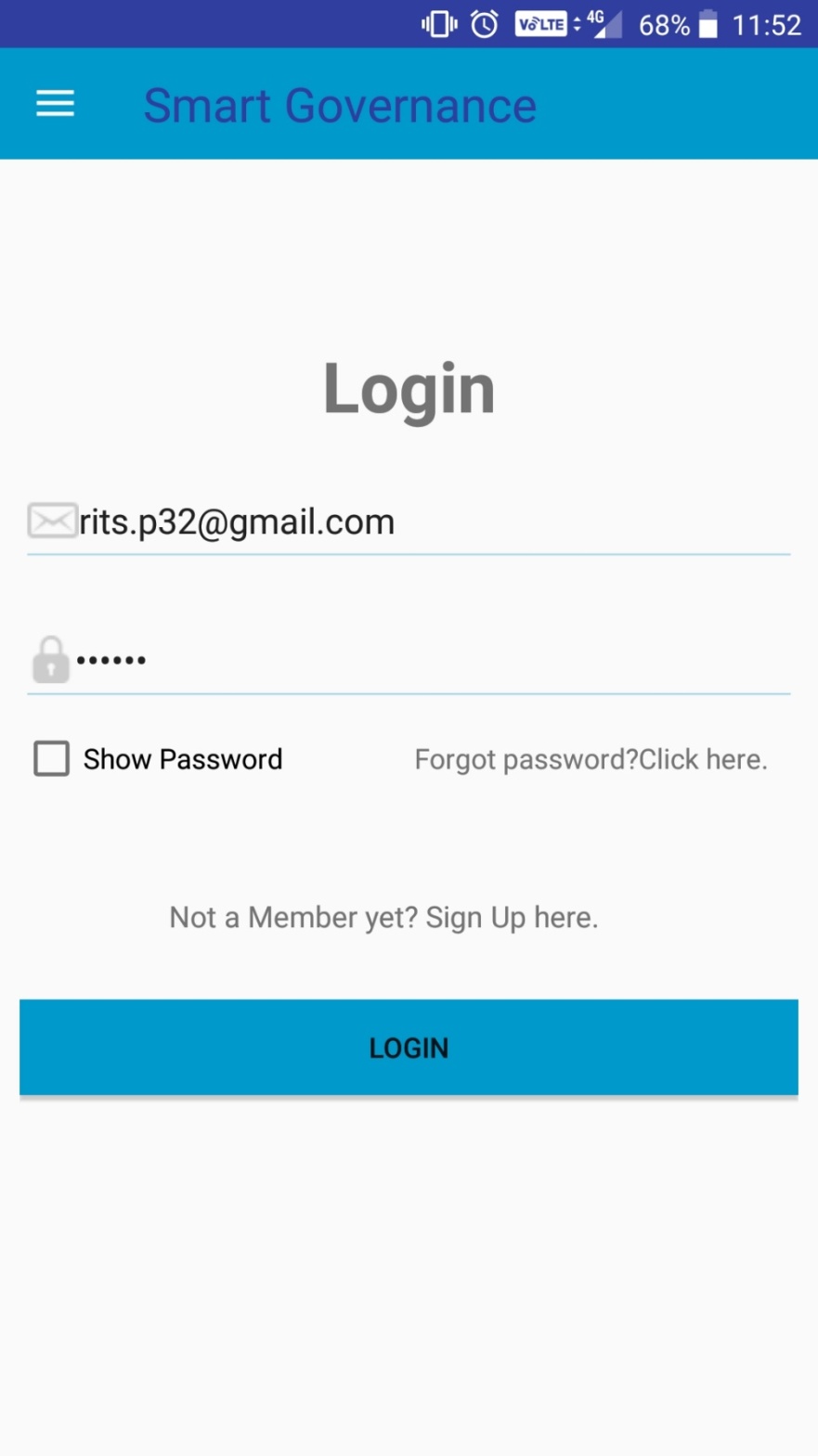


10. HOW TO USE APPLICATION:



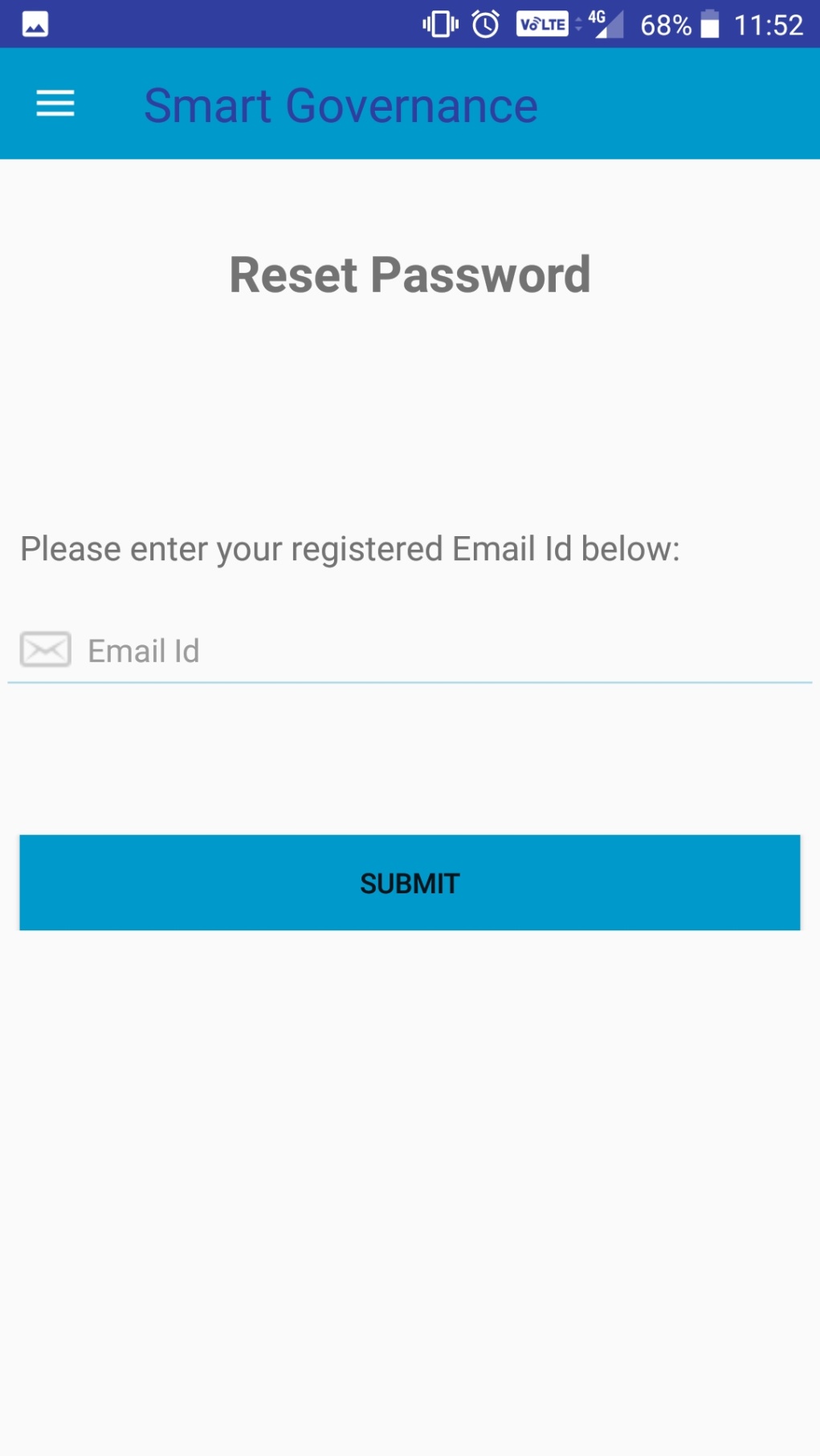
Login Page

This page is used to enter the login details viz. email and password. Also, a link for forgot password and to sign up is there.



Reset Password/Forget Password

This page is used to enter the email id wherein the link to reset the password is mentioned.



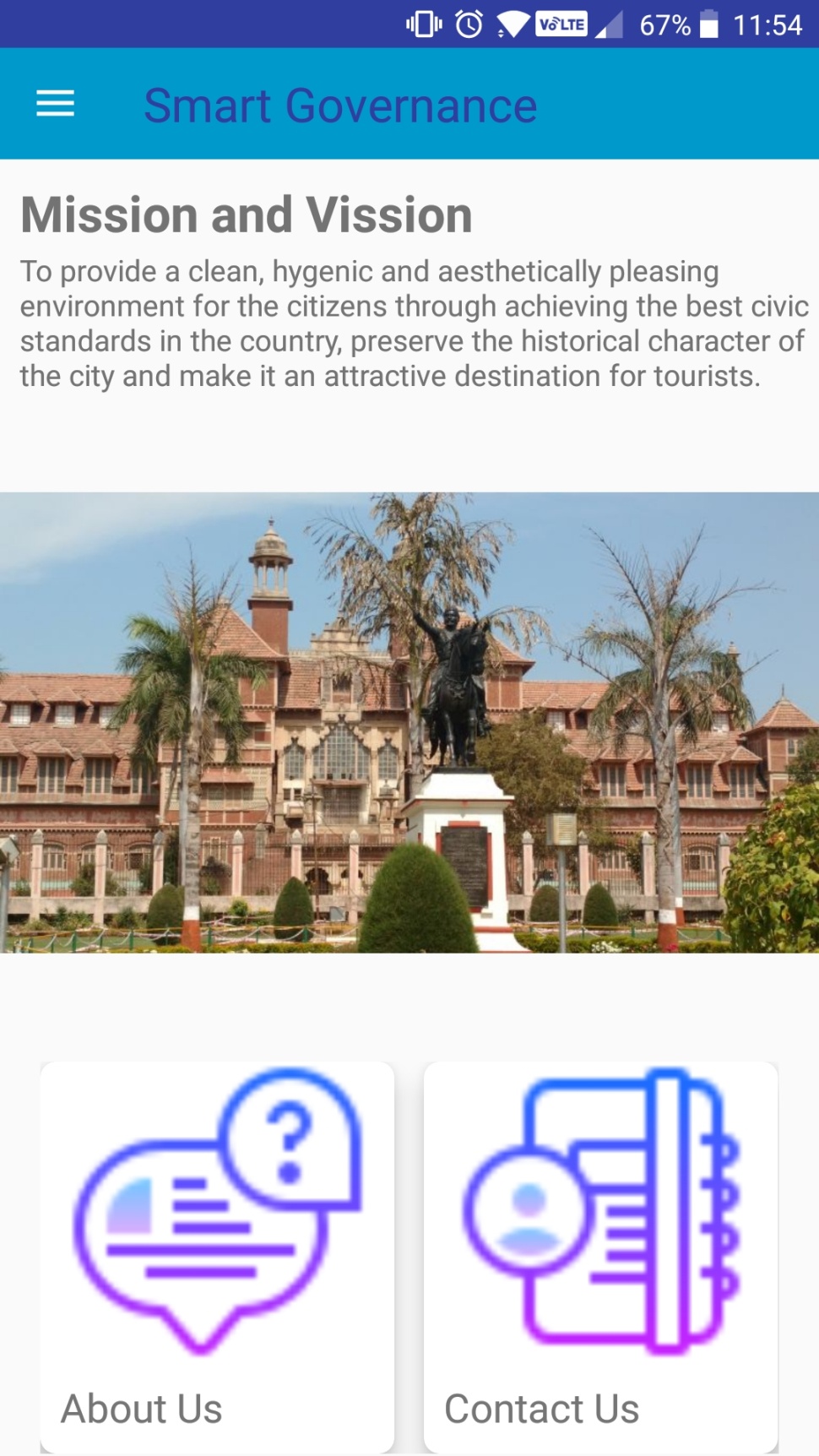
Sign Up

Sign Up page to register as a new user.



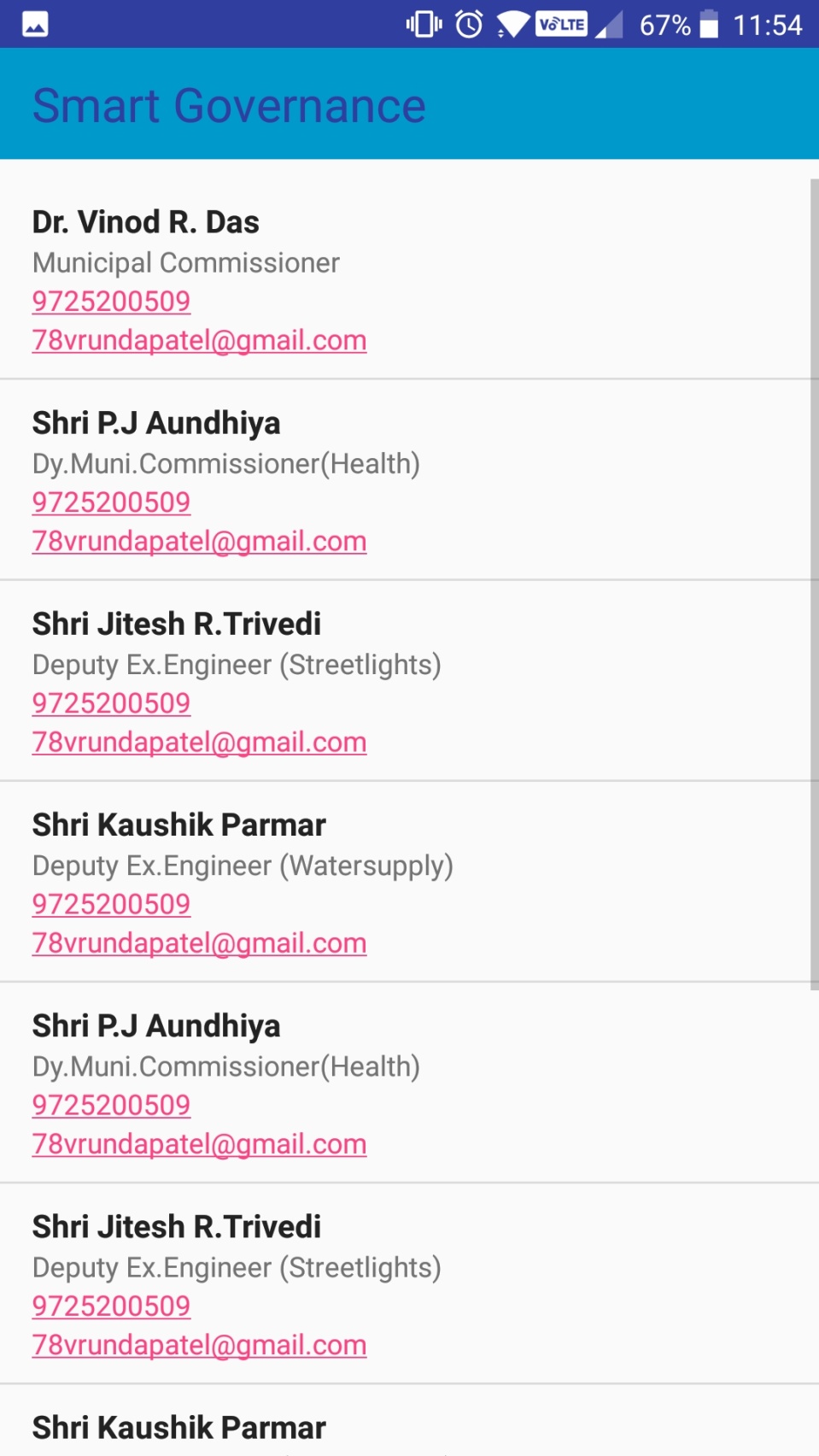
Home Page

This is the first page that the user sees when he/she is logged in. It contains the vision and mission and links to about us and contact us.



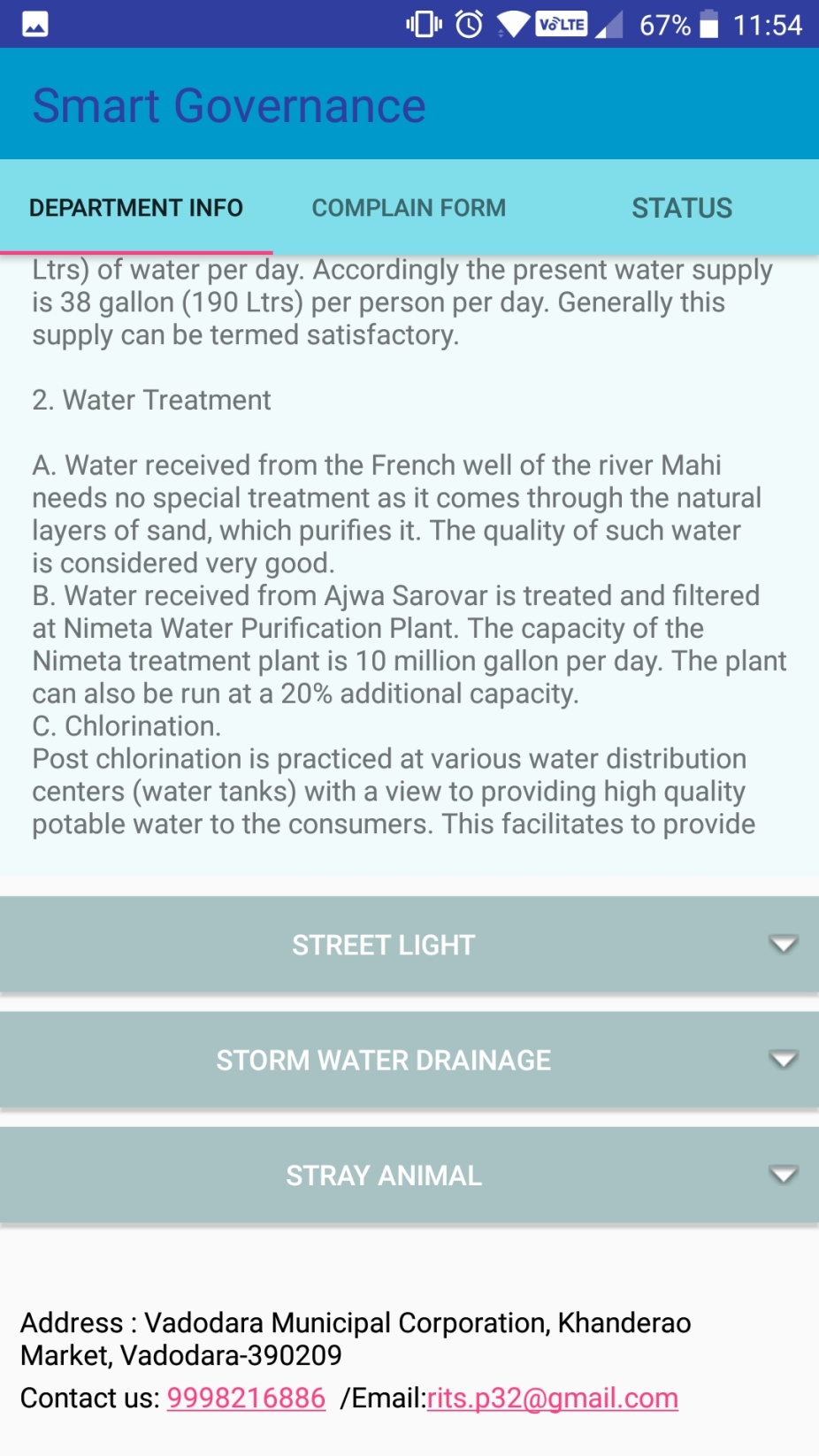
Contact Us

Contains the contact details viz. name, designation, contact number and email id.



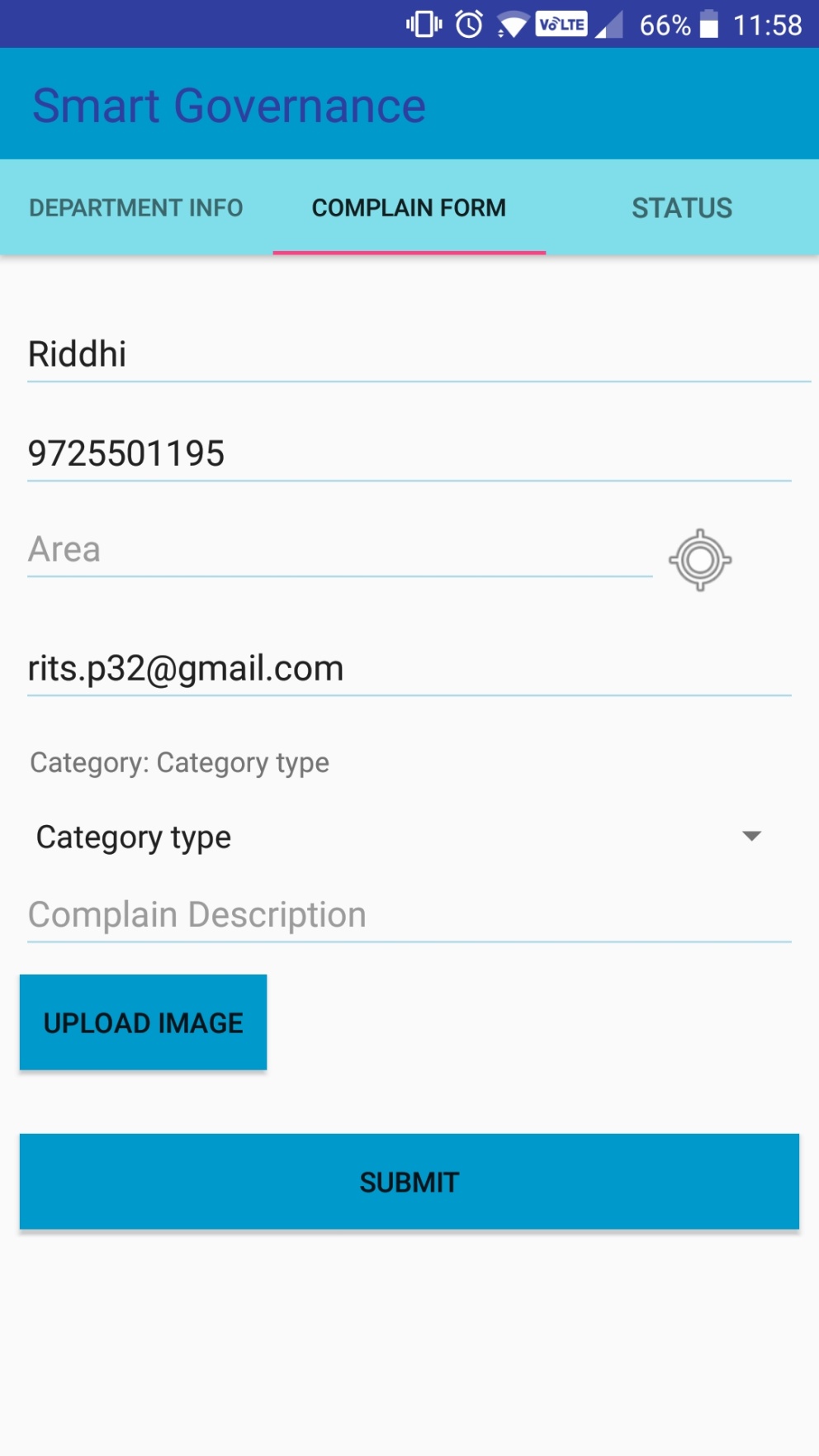
Departments

This contains department info, complain form and status of complaint.



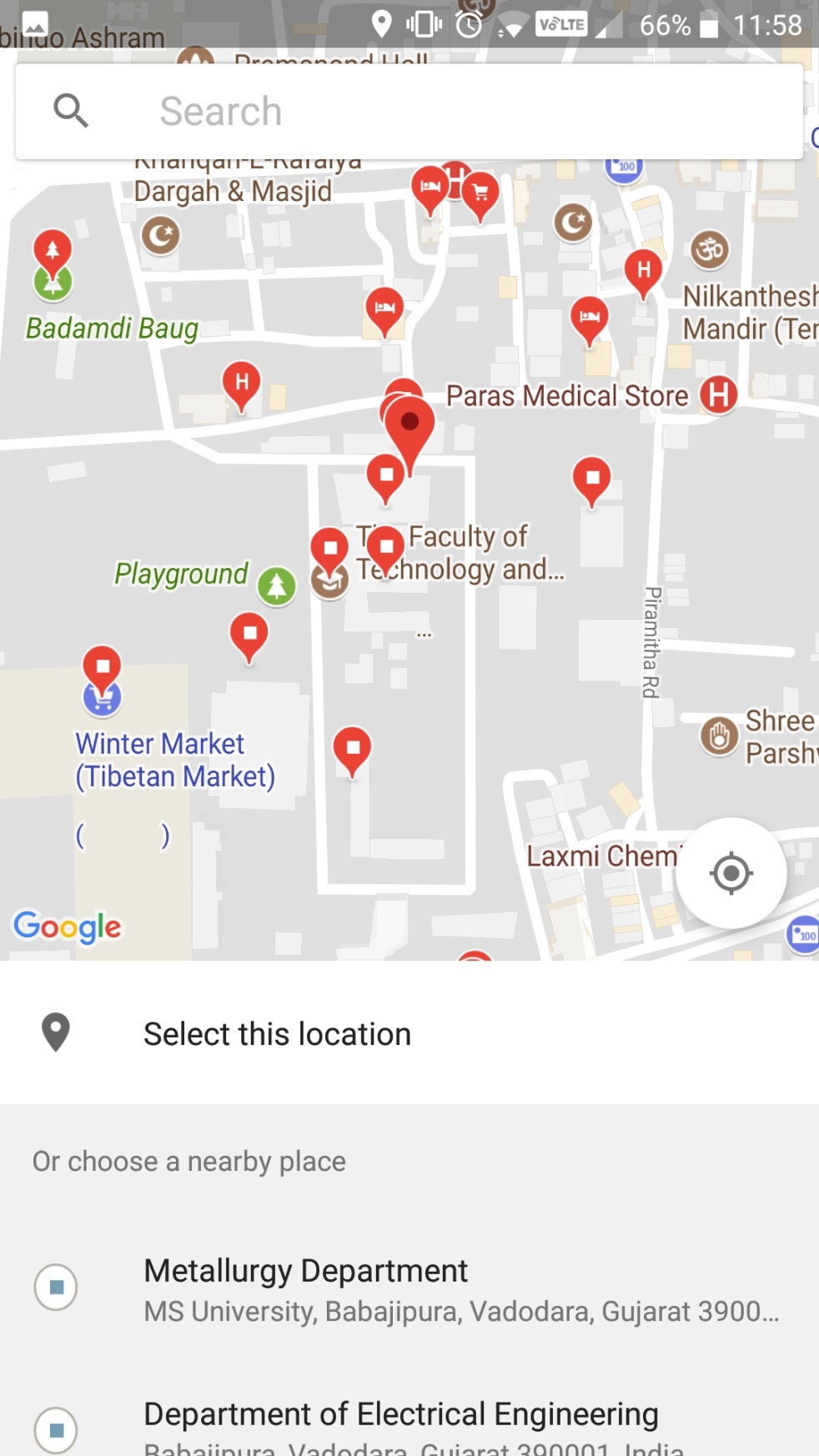
Complaint Form

This is the form to give a complaint regarding a certain department, where images can also be uploaded.



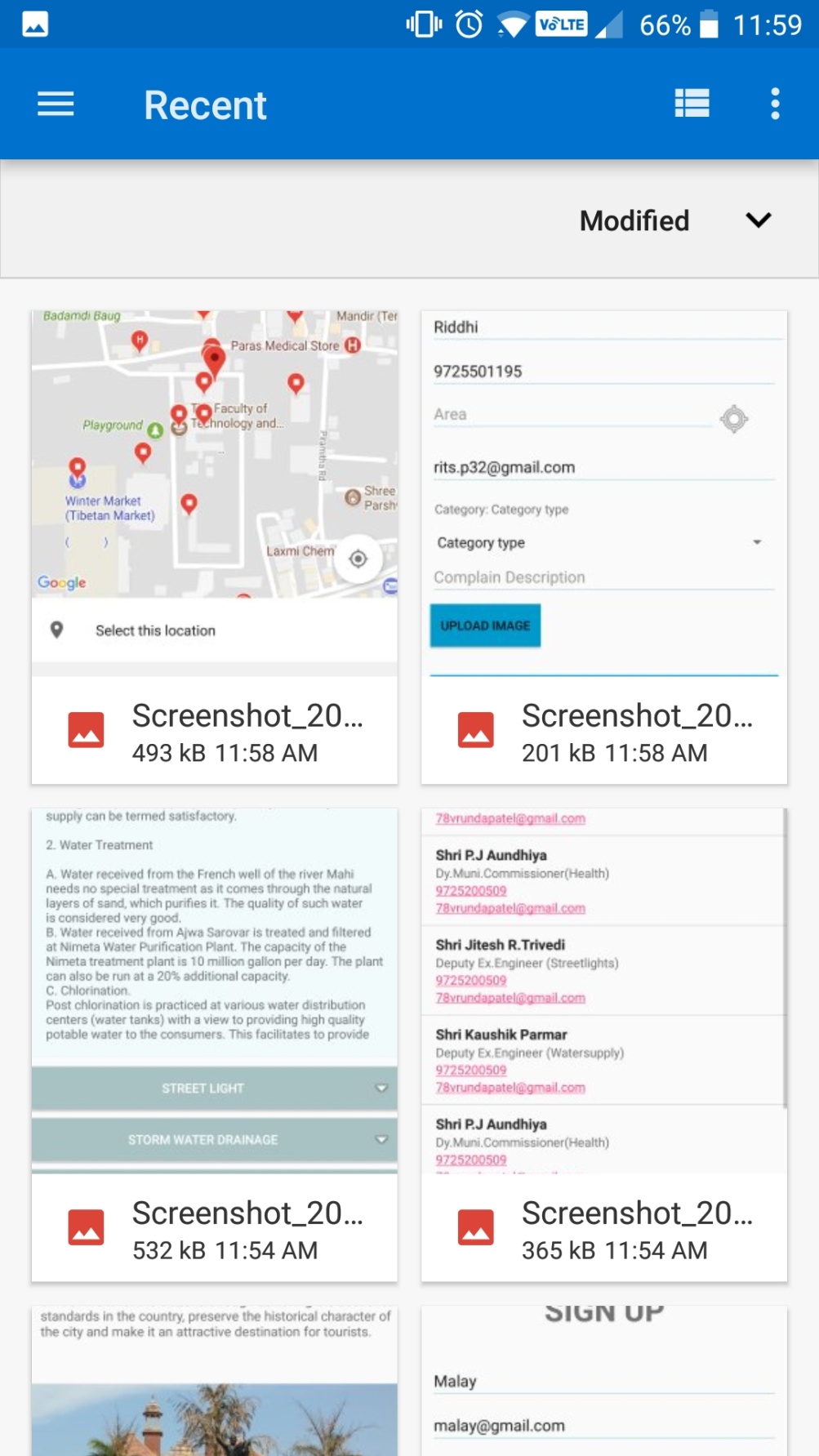
Fetching Location

This is used to fetch the location of user when the icon next to the Area is clicked. It fetches the location from Google Maps.



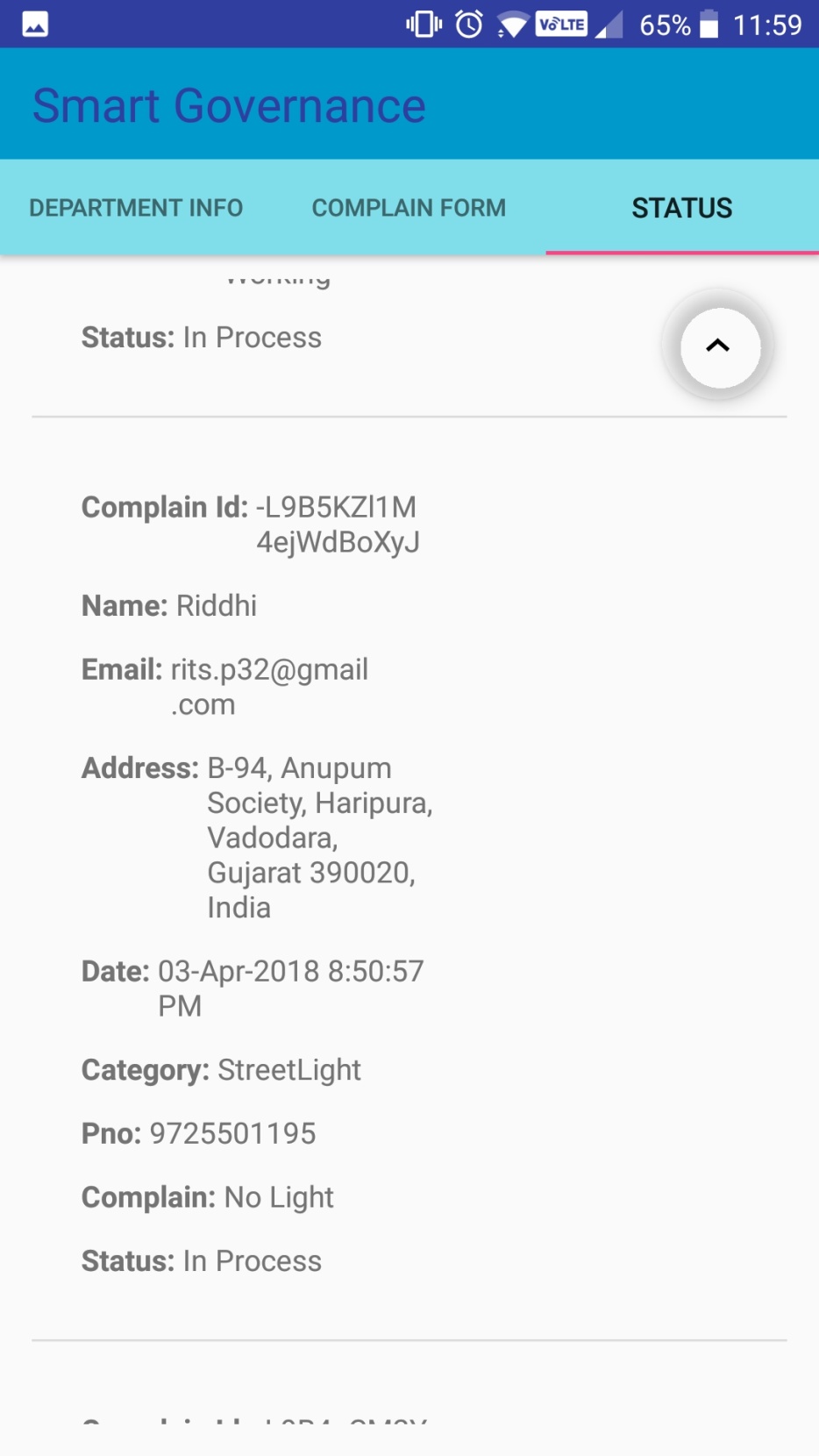
Fetching Images from Gallery

In the complaint form, the images can be fetched from the gallery of the device.



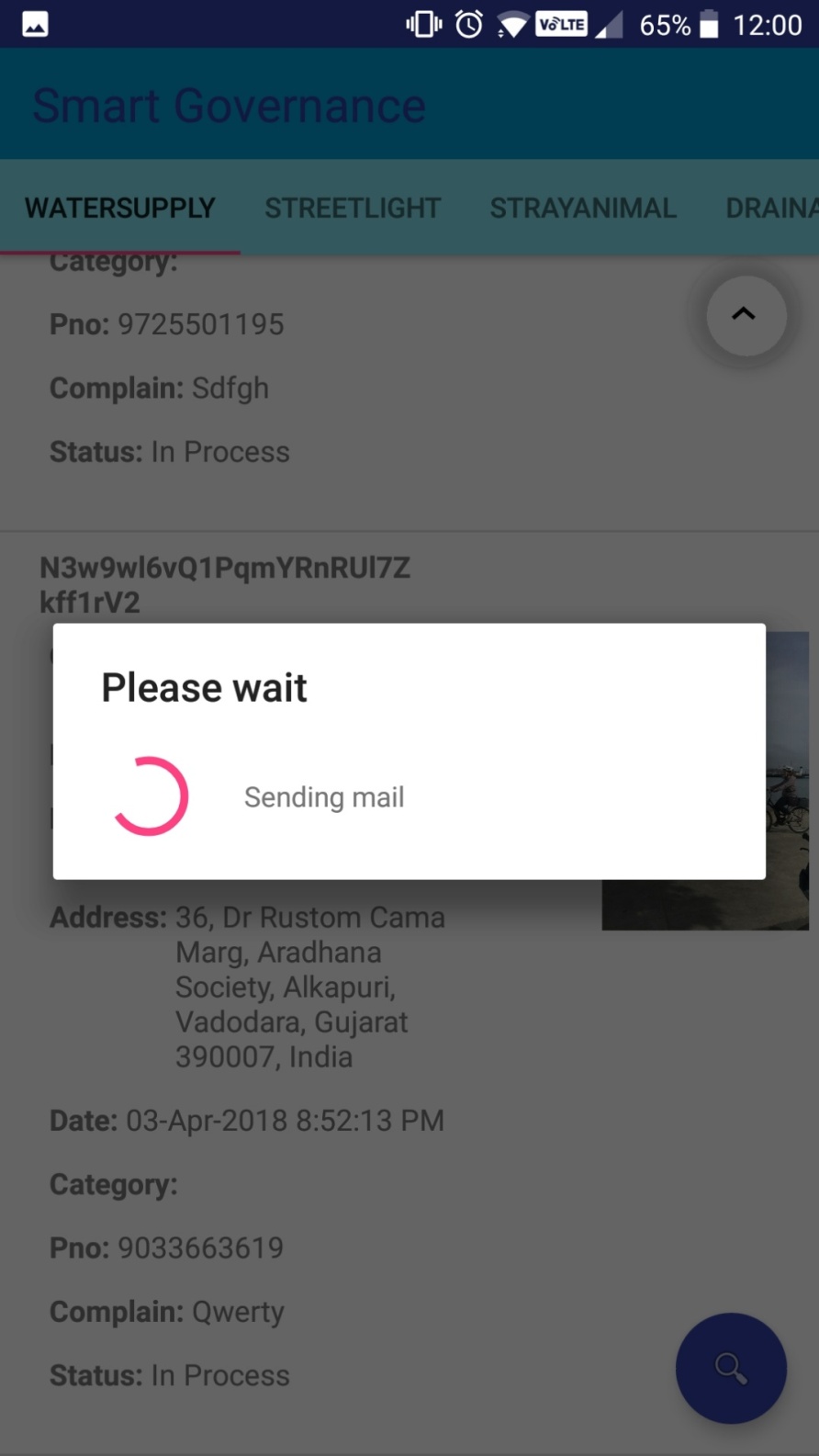
Complaint Status

The status of all the complaints entered by the user is mentioned here, and their status is also mentioned.



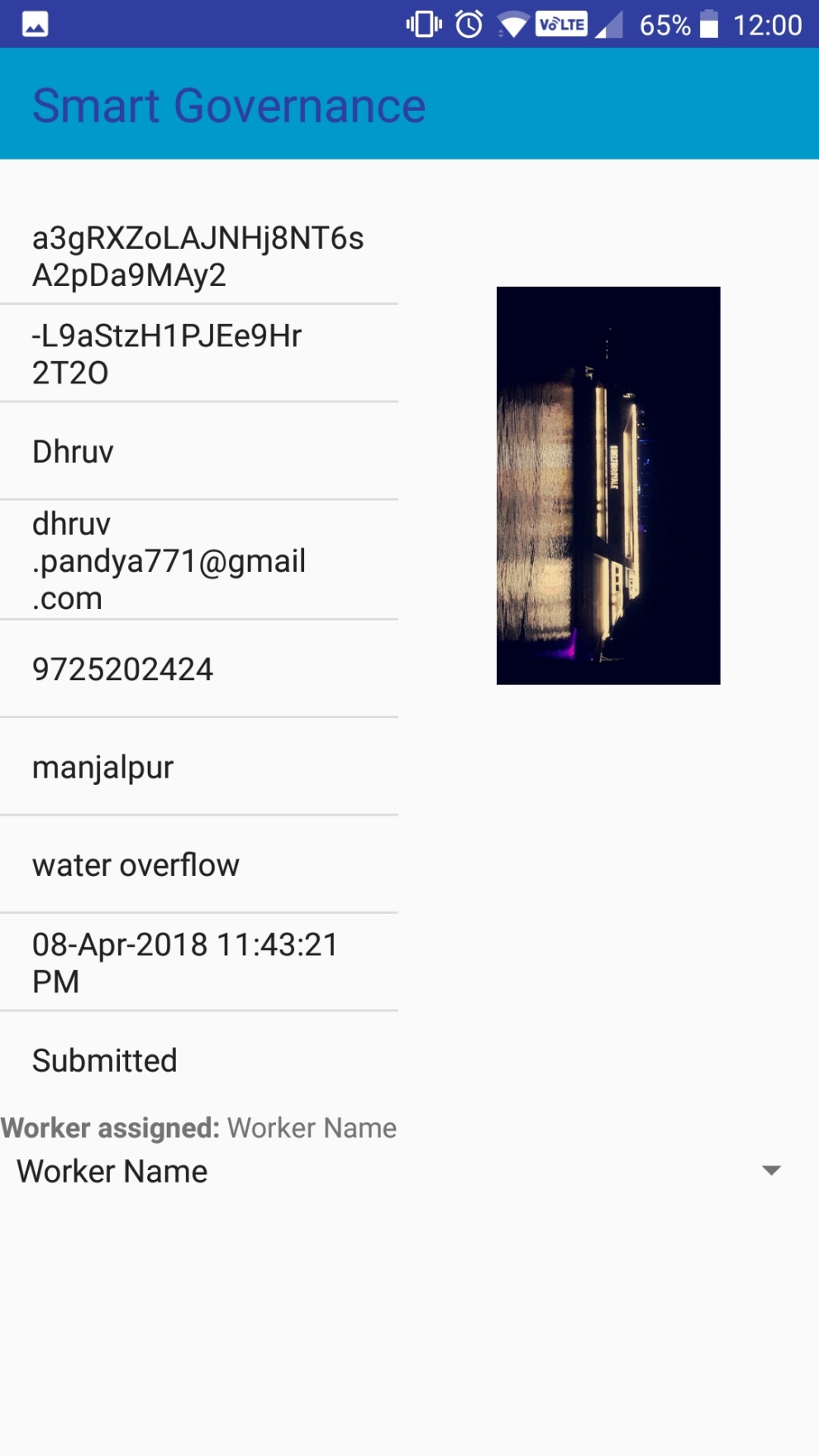
Reminder Mail

2 days before the due date for completion of job, if the job is not done, then a reminder email is sent to the concerned worker automatically.



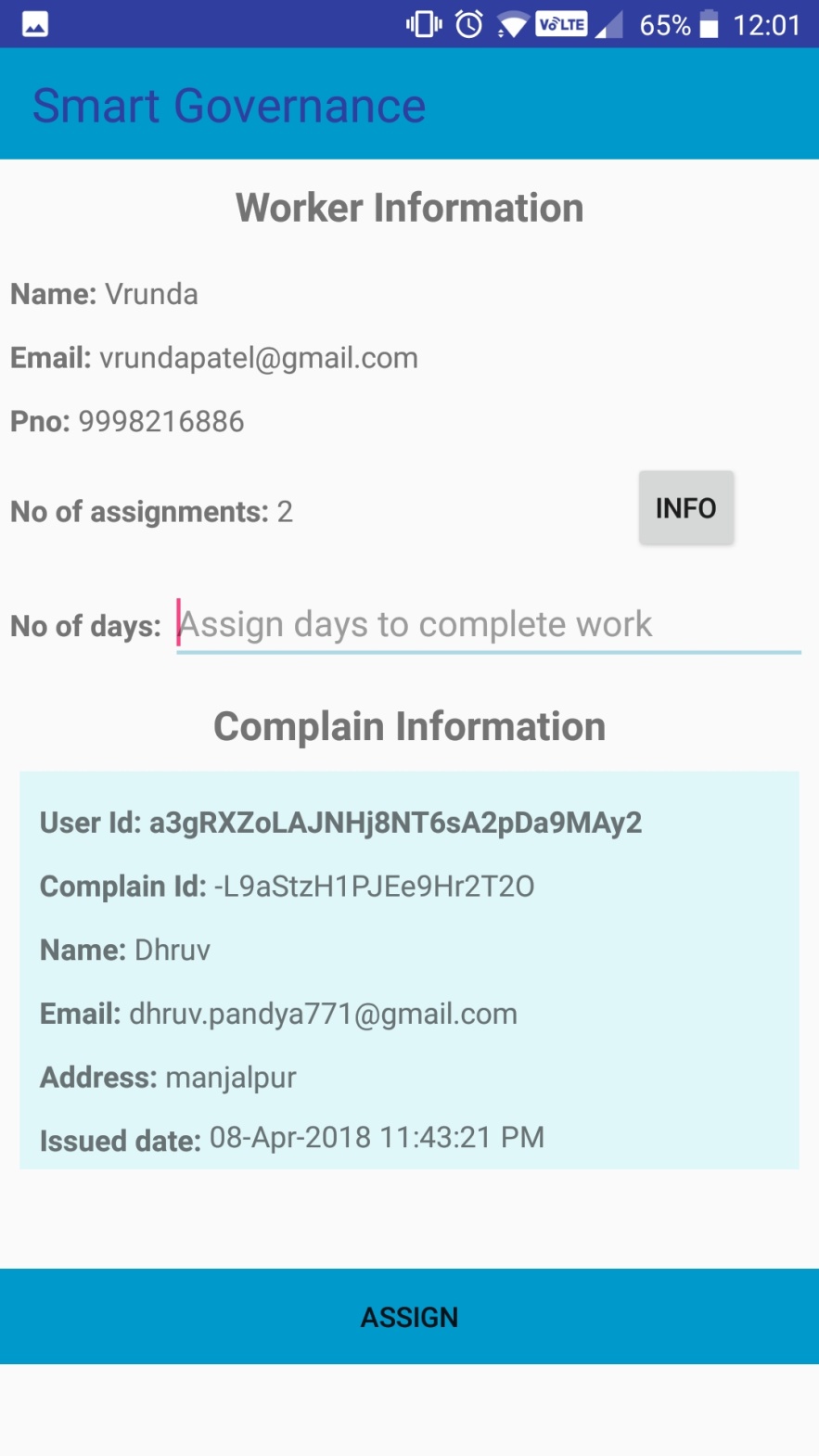
Assigning a complaint to Worker by Admin

When a worker submits the complaint, the admin can assign a worker via the drop down list to solve the concerned complaint.



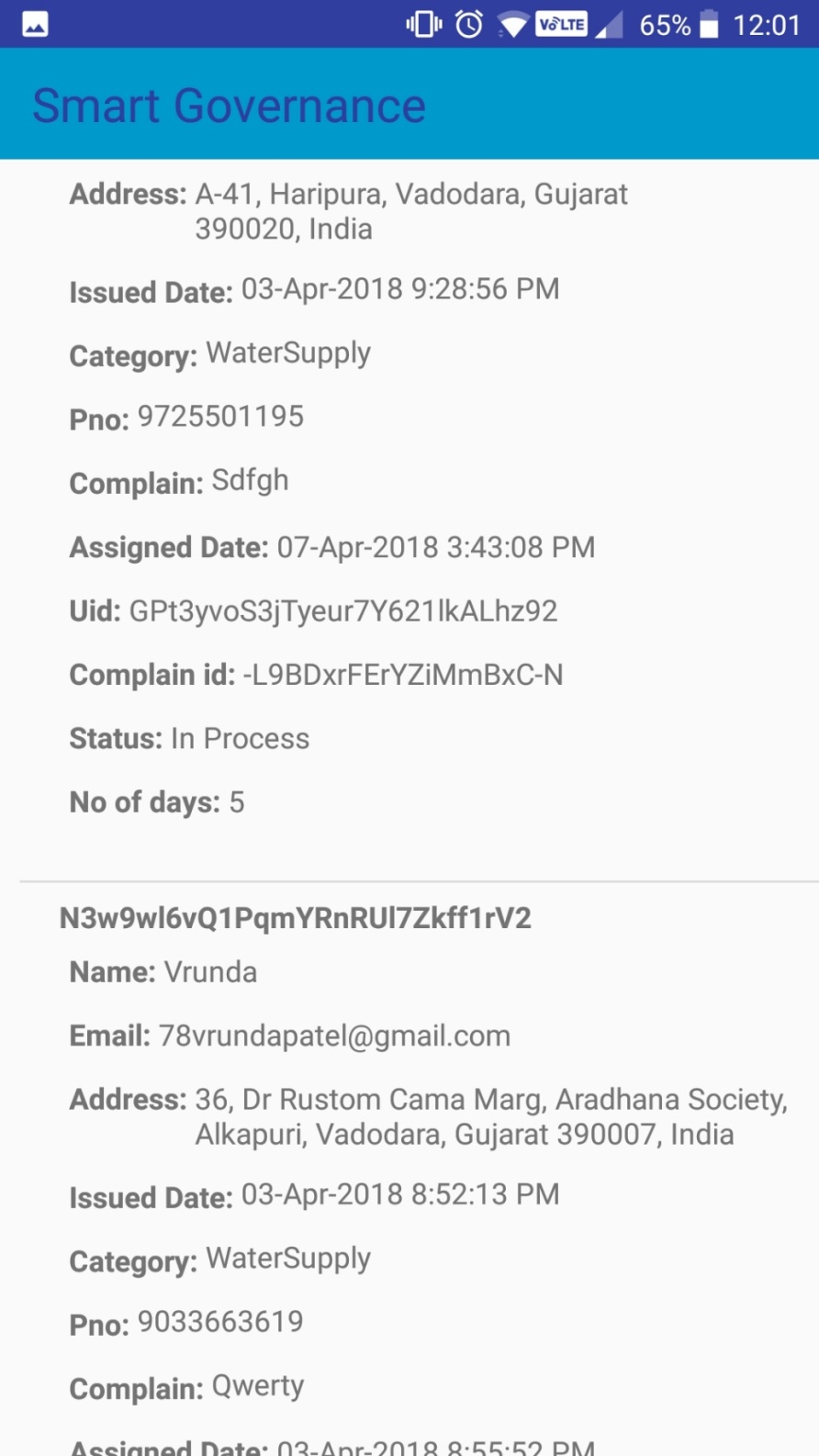
Assigning number of days to solve the complaint to the Worker by Admin

The worker assigned for solving a particular complaint can be assigned number of days to solve the same via this page by the admin.



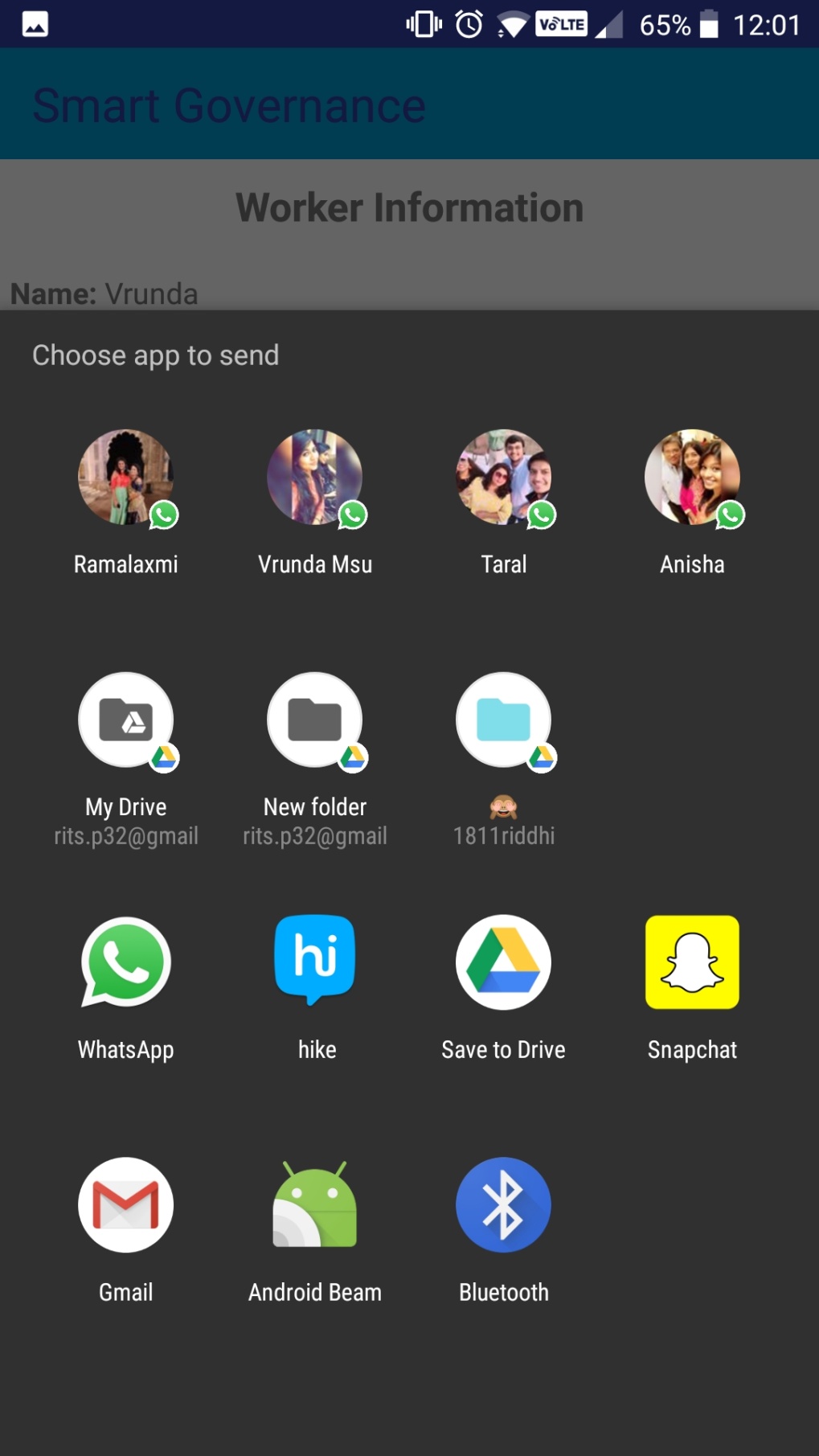
Complaint Information

This page is accessible only by the Admin and it contains all the complaints details and their status and the workers assigned for the complaint and the number of days remaining to solve the same.



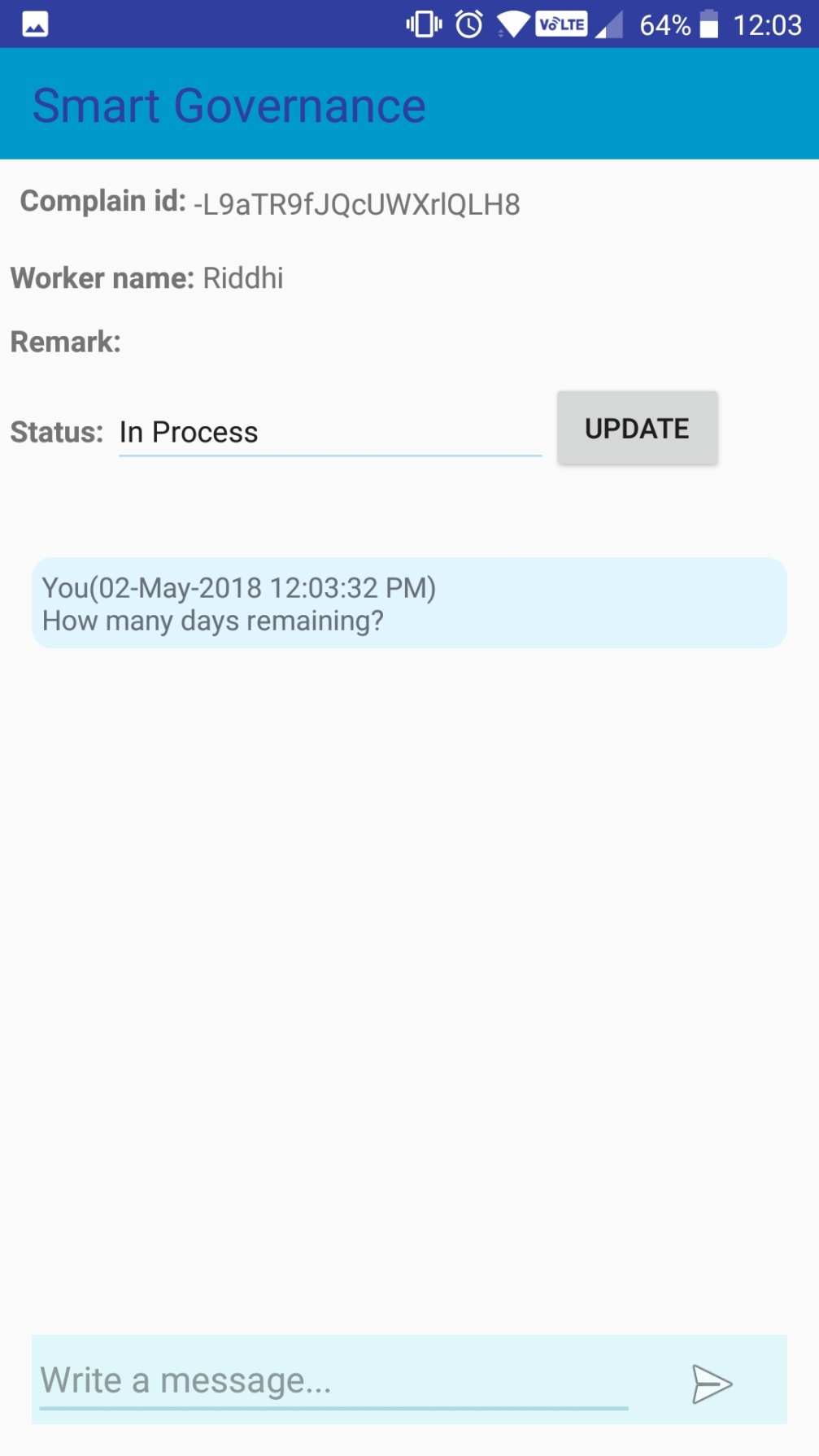
Send mail to the worker by Admin

This is used by Admin to send the details of the complaint to the assigned worker.



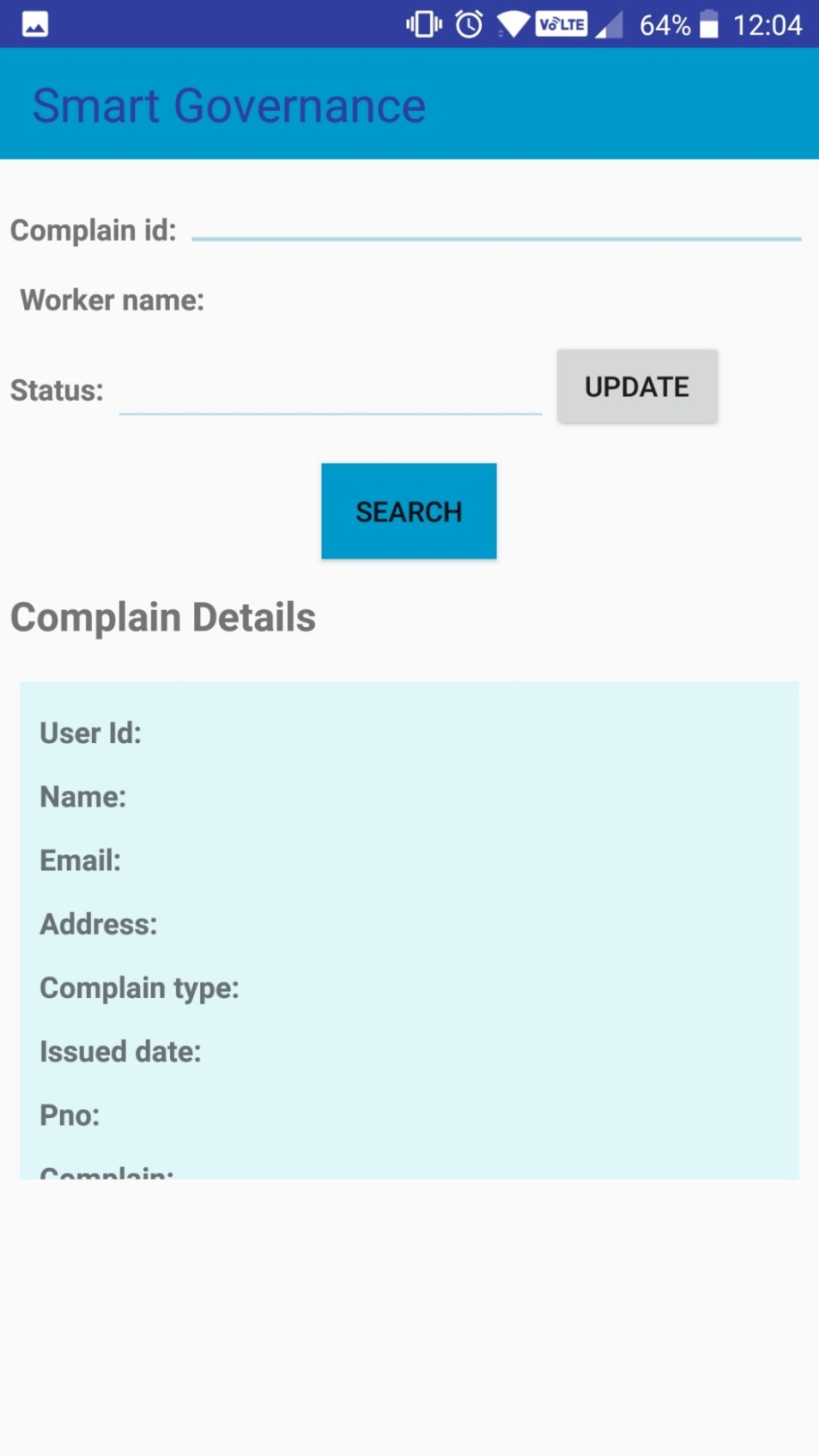
Chat page between Admin and assigned Worker

It’s the page on where the assigned worker informs the admin about the progress of the work , and also the admin can update the status of the complaint. Only admin can view this.



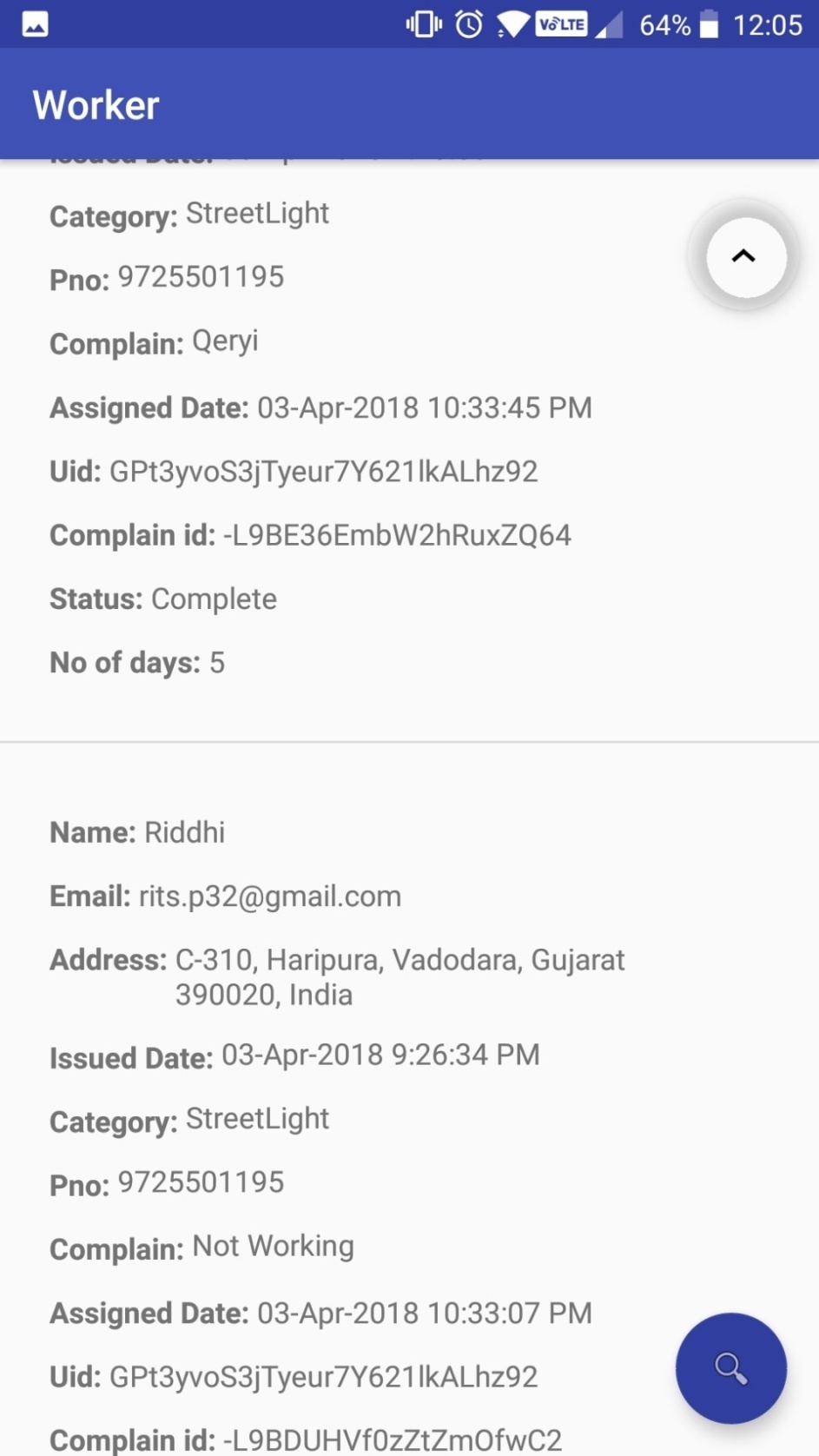
Searching for complaints by Admin

The admin can search the complaints via the complaint ID and the admin can change the status of the same too.



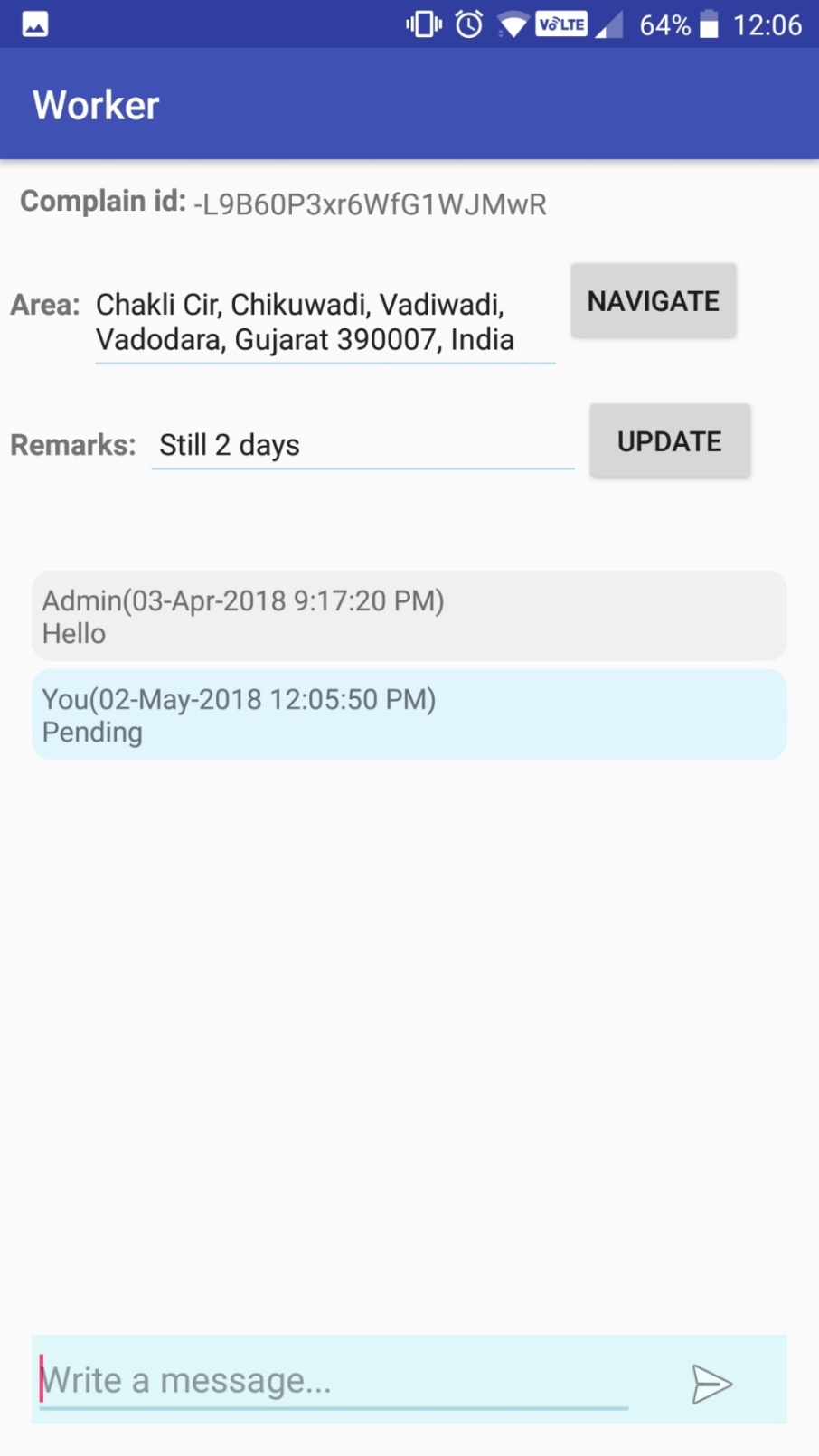
Worker Page

This page is accessible only by the worker and it shows how many complaints the worker has been assigned across different departments.



Worker side of Admin Worker Chat

Here, the worker can send messages that only admin can see and worker can also add remarks which can only be seen by the admin.



Navigation for Worker

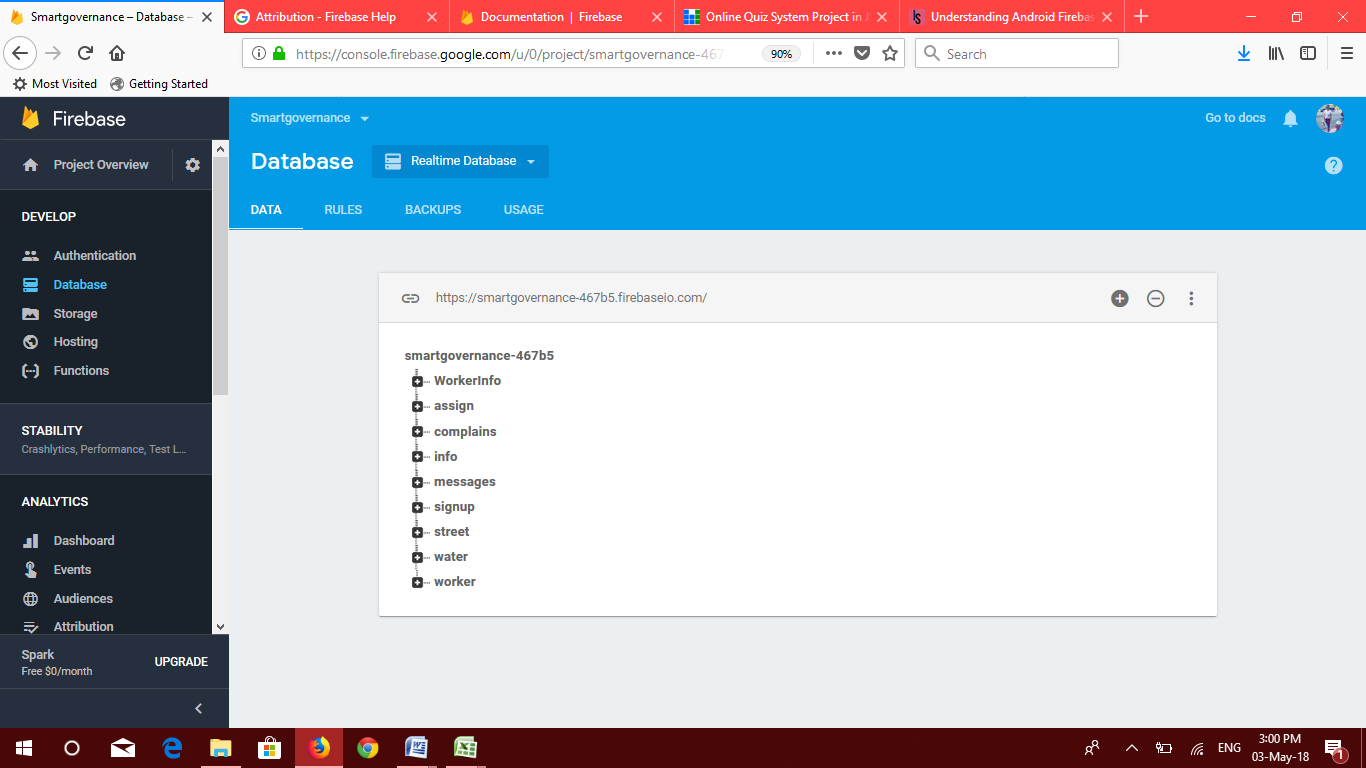
This is the navigation screen for the worker to the area of the complaint, via Google Maps.

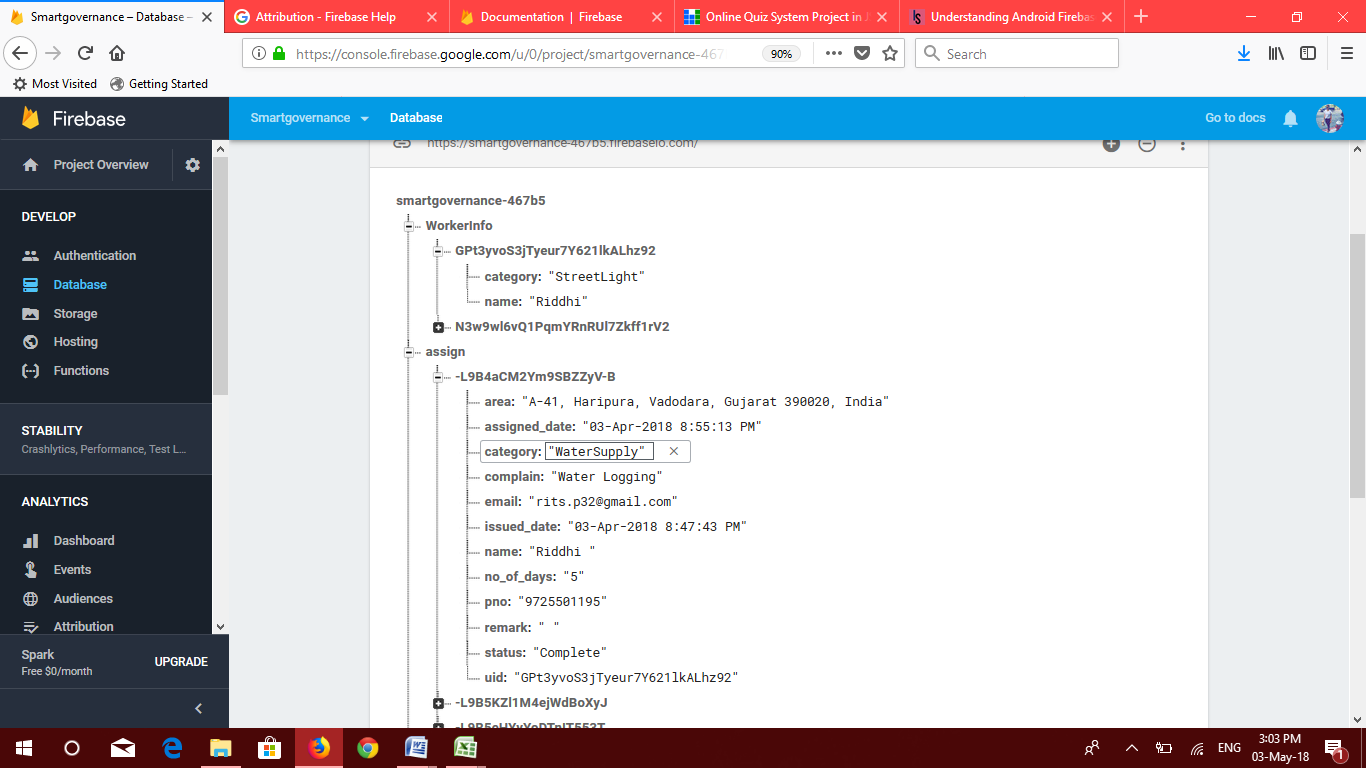


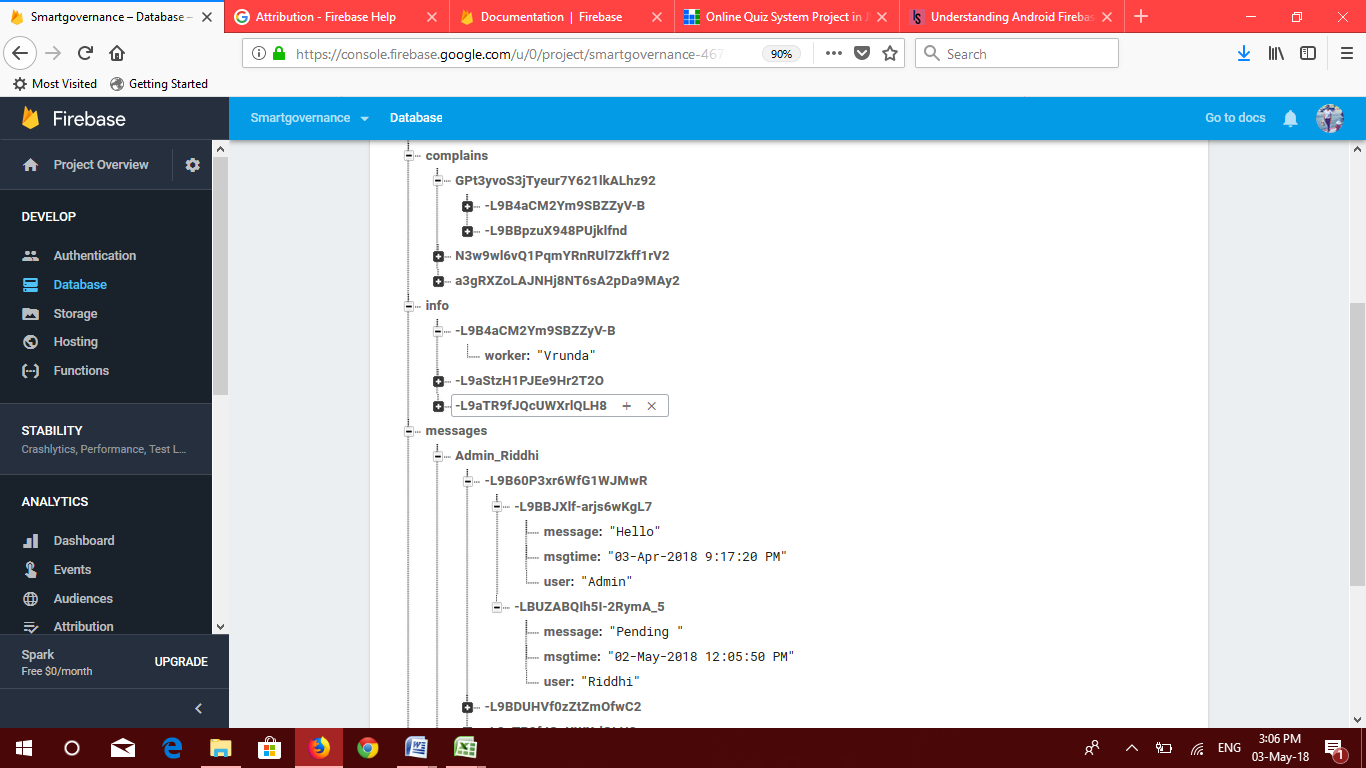
11. DATA DICTIONARY:

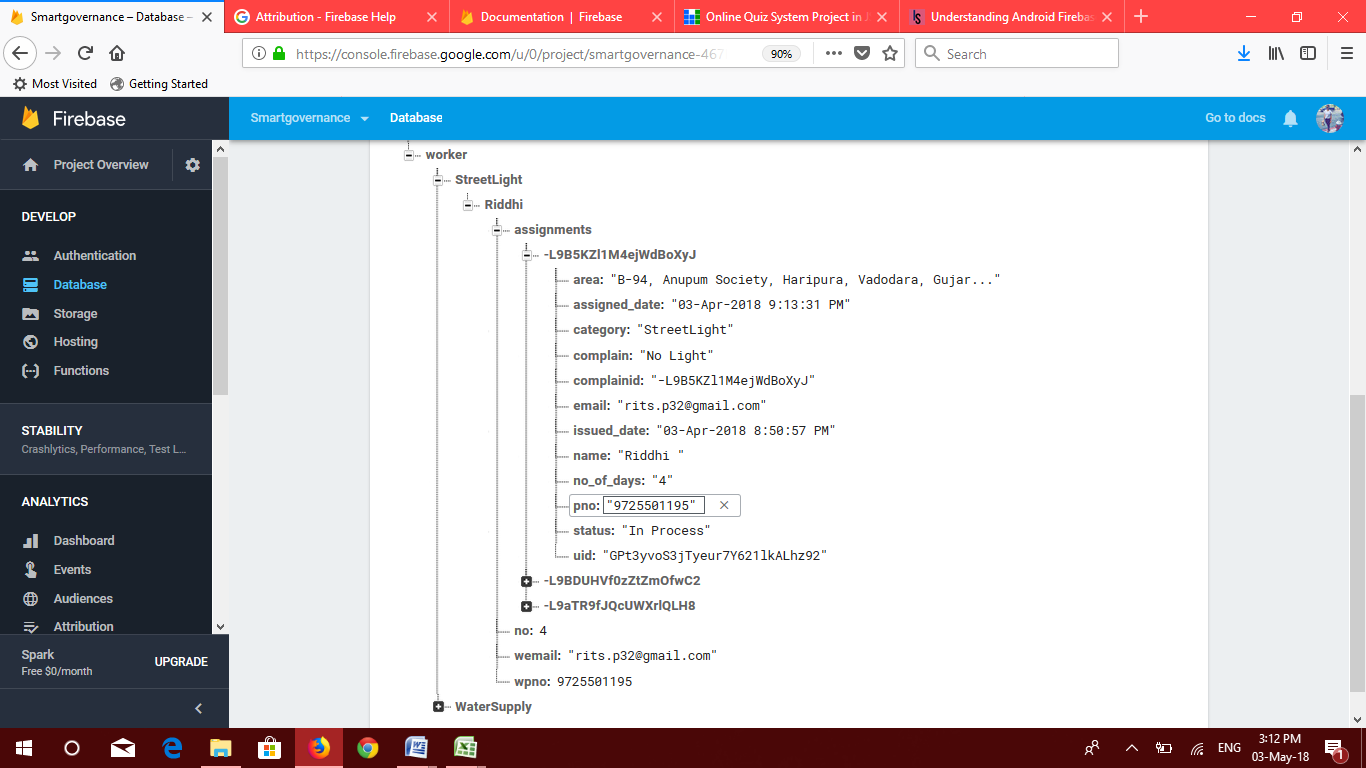


|  |  |
| --- | --- |
| Tables | Attributes |
| WorkerInfo | category, name |
| Assign | area, assigned\_date, category, complain, email, issued\_date, name, no\_of\_days, pno, remark, status, uid |
| Complains | area, category, complain, complain id, date, email, imageURL, name, pno, status |
| Info | complainid, worker |
| Messages | message, msgtime, user |
| Signup | Address, Email, Name, Pno, UserId |
| Departments | area, complain, complain id, date, email, imageURL, name, pno, status, uid |
| Worker | Department\_name |
| Department\_name | woker\_name |
| worker\_name | wmail,wpno,assignments |
| assignments | area, assigned\_date, category, complain, complainid, email, issued\_date, name, no\_of\_days, pno, status, uid, no, wemail, wpno |









12. APPLICATION ANALYSIS:



What can we track?

Event properties

The firebase SDK provides a collection of commonly used events in the form of predefined constants that can be used when tracking events. If you are only performing simple tracking, then the predefined events should cover your needs. On the other hand, using custom event names allows you to track events that are specific to your app and will allow for a deeper analytical understanding when it comes to the reports created from your tracking.We can track an event without sending any parameters by simply calling the logEvent() method with the event name.

Syntax:

Bundle bundle = new Bundle();

bundle.putString(“item\_purchased”, “Pizza”);

bundle.putInt(“item\_quantity”, 1);

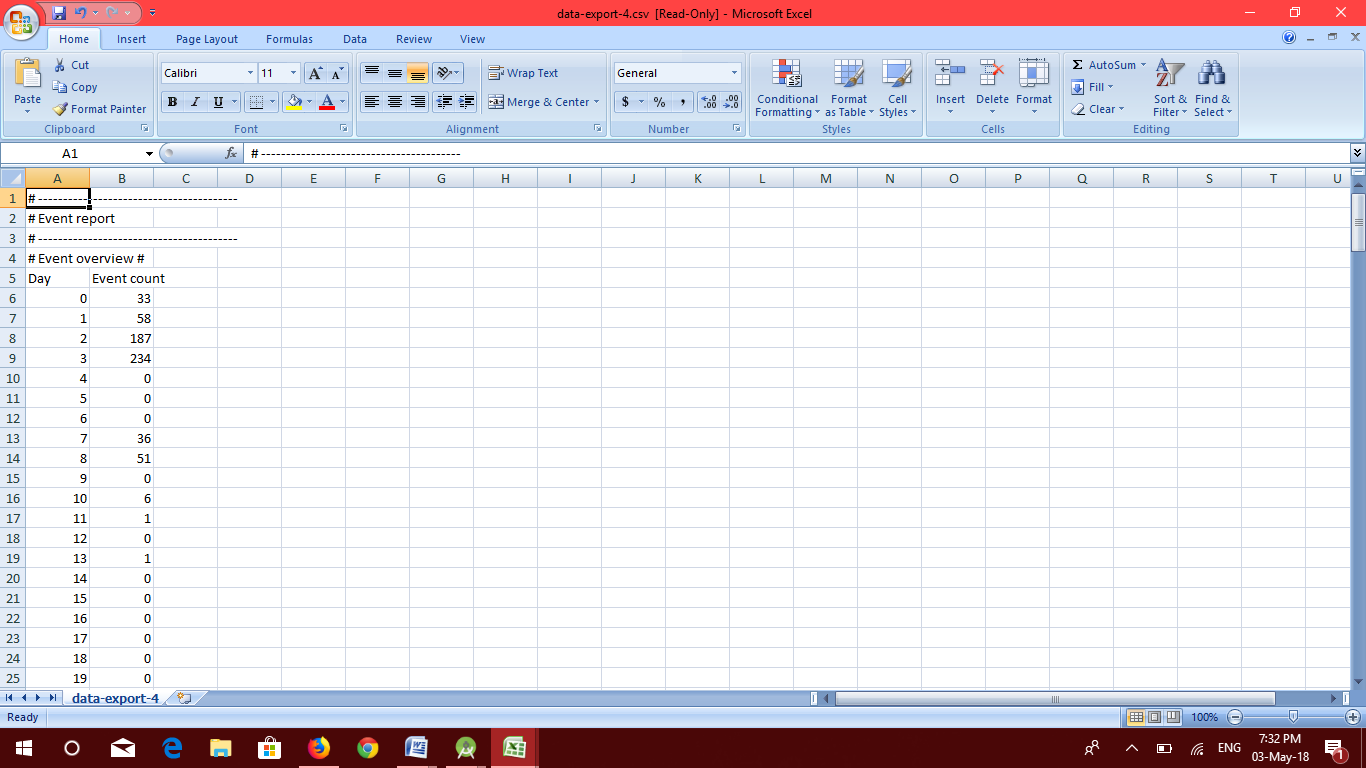
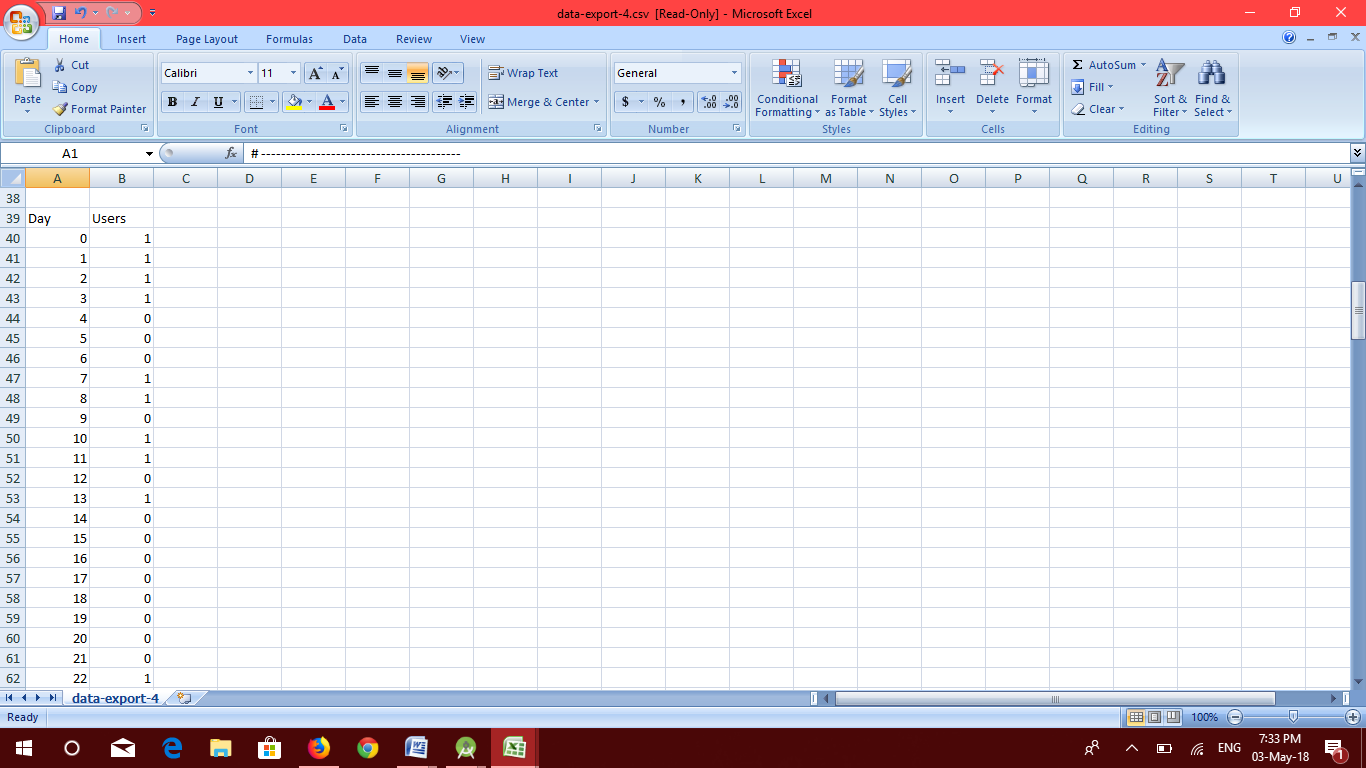
firebaseAnalytics.logEvent(“checkout\_complete”, bundle);

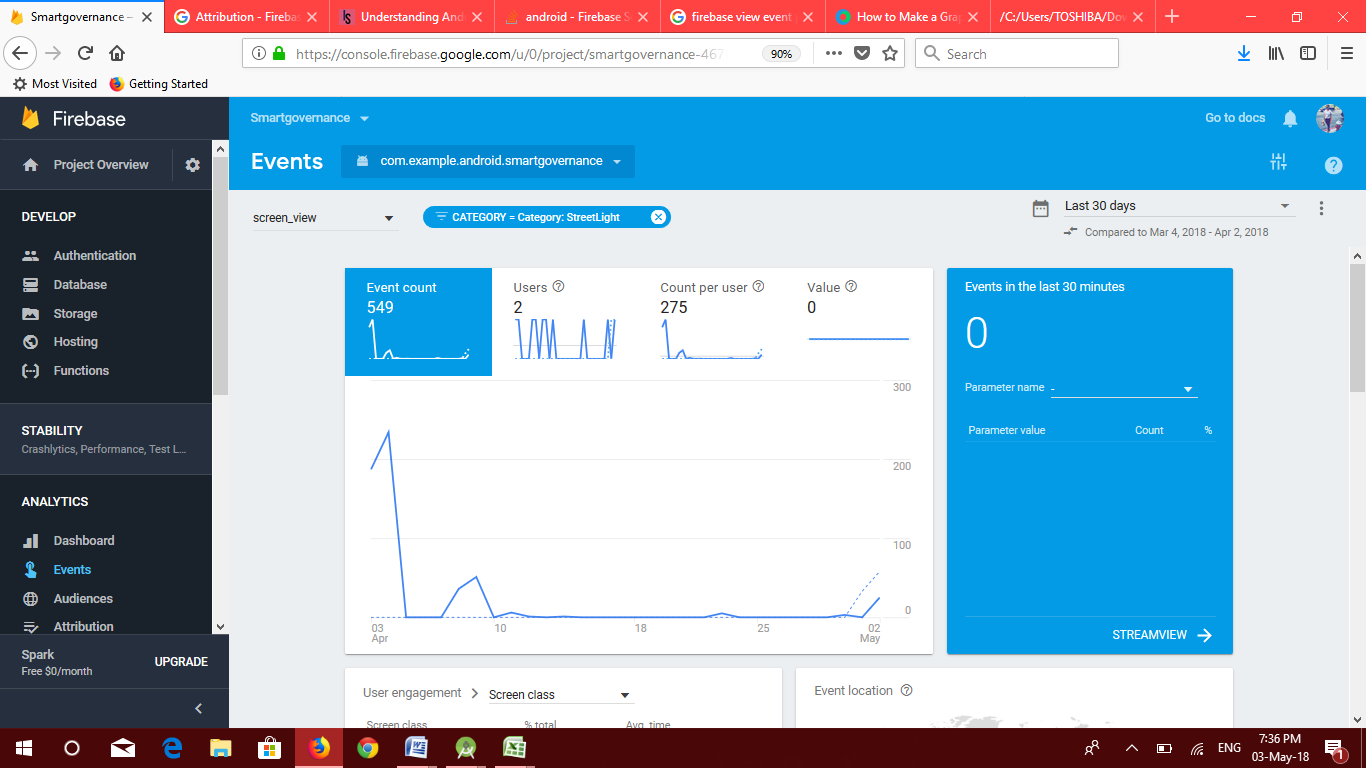
User Properties

User properties allow us to track data that is specific to the users who are engaging with our app. This allows us to keep a track of data that is not specifically associated with our application itself, but more focused on the users.

Syntax:

firebaseAnalytics.setUserProperty(  
 “favourite\_film\_genre”, filmGenre);



13. FUTURE ENHANCEMENTS:



* With good expert advice the application can be made available to users on the Play store.
* We can add the other functionalities related to the Municipality office like E-chalan.
* The work of the admin can be made automatic by using AI while assigning the complains to the workers.
* Better error handling in case of network failure .

14. BIBLIOGRAPHY:



List of the references used are:

* <https://www.tutorialspoint.com>
* <https://www.javacodegeeks.com>
* <https://www.androhive.com>
* <https://www.stackoverflow.com>
* <https://firebase.google.com/>
* <https://developers.google.com>
* Professional Android Application Development by Wrox publication