

# E-Governance via Online and Offline Server

Arjit Madan<sup>1</sup>, Ayush Sharma<sup>2</sup>, Krishna Bihari Dubey<sup>3</sup>

<sup>1</sup>Research Student

<sup>2</sup>Research Student

<sup>3</sup>Assistant Professor, ABES Institute of Technology, Ghaziabad, India

\*\*\*

**Abstract** - This Project is insight into the design and implementation of developing an application for providing Government Services. The app is designed with an easy to use GUI (Graphical user interface). Users can sign up and verify them-selves using Email id and then can login using the same. A dashboard of services will be displayed from which the user can choose and then it can provide with necessary details on the portal. Example: We have chosen emergency services which are not provided at a faster pace due to unavailability of efficient communication services. So we have provided online as well as offline facility

**Key Words:** GUI, E-mail, Android, SMS, Database, Server

## 1. INTRODUCTION

E-Governance is the need of the hour and many citizen centric services are either online or going to be online. Any citizen to take benefit of these E-Governance services need to have an access to any computer/mobile and a good internet connection. But citizens are not able to take full benefits of these citizen centric Government services because of Internet connectivity issue. Even at some locations there is no internet connection at some times. As most of the citizen services' portals are using text for communication and therefore this application provides a service which has an easy to use GUI and in the background it normally uses internet services and in case the services are not available then automatically it uses encrypted SMS to request data from server and the server is able to send the data to the user mobile app via encrypted SMS which the app decrypts and provide on the GUI of mobile app.

## 2. LITERARY SURVEY

### 2.1. Related work

A lot of research work has been done on E-governance applications and their offline working but analysing these papers all I found out that they did not work when there was no internet connection, and that there should be easy to use GUI(Graphical User Interface), it should use strong password policies to reduce hacking by using encryption. The data which is also to be sent should be encrypted. Today, Government of India is providing a lot of services to the citizens to make their daily life easy and secure.

But due to lack of information a large portion of the population is not able to take advantage of these services. Although most of the people are using smartphones but internet connectivity is still not available everywhere. Therefore, we have come up with a mobile application which collectively provides the citizens with necessary government services either online or offline (via cellular connection). There is an app named "Umang". Umang is the first-ever solution to provide Government Services online. This app provides services like Ayushman Bharat, Pradhan Mantri Awas Yojna, Jan Dhan Yojna and many others. Since the internet has the power to provide amount of information to the world, it is also utilized to provide services sitting in comfort at your home with services a tap away.

### 2.2. Proposed solution

The app is designed with an easy to use GUI (Graphical user interface). Users can sign up and verify them-selves using Email id and then can login using the same. A dashboard of services will be displayed from which the user can choose and then it can provide with necessary details on the portal. Example: We have chosen emergency services which are not provided at a faster pace due to unavailability of efficient communication services. So we have provided online as well as offline facility.

Offline portal:

When there is no internet connection the app will switch to offline mode. The information is sent from the user using mobile carrier's SMS service. The SMS is sent in an encrypted form. Here we are using AES encryption algorithm. Encrypted SMS is sent from the server and decrypted at the android side and vice versa.

Offline Server:

Server receives SMS from the user using TextLocalSms API. Then the message will be decrypted by the JAVA application using javax.crypto package. If the user has requested for the service, then forwarded to the concerned department or else if it is information required then it can be fetched from the database.

Department/Administration:

The department receives the request from the server on the administration portal. Administration responds accordingly

back to the server from where the information (form of encrypted SMS) is sent back to the user where message is decrypted and is displayed on the GUI of the app.

### 3. WORK FLOW

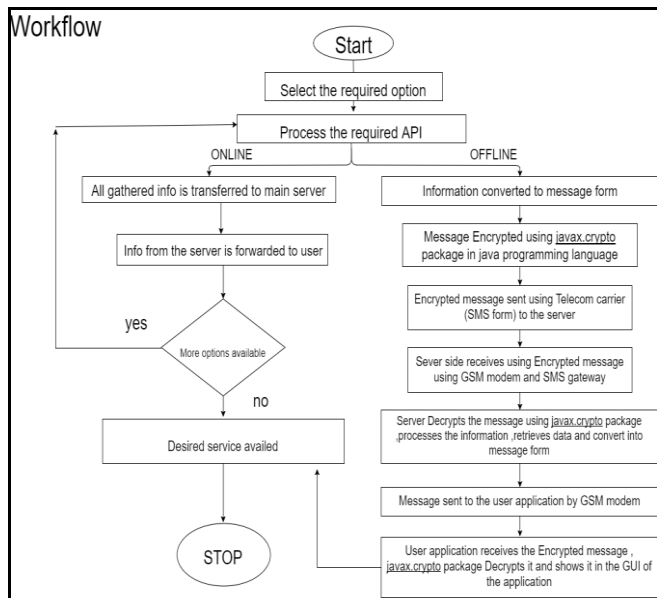


Fig -1: Workflow

1. User has to select the desired option.
2. Upon selecting the desired option, an associated API is processed.
3. After this the checks whether there is internet connectivity or not and based on the result, the application switches to the relevant mode.
4. In online mode, every information that has been entered, is sent to the main server.
5. The Server processes accordingly and sends back the data to the user via internet.
6. In offline mode, information is converted into an encrypted message.
7. The encrypted message is then sent using the telecom carrier SMS service, to the server.
8. Server receives the encrypted message using SMS gateway.
9. Server decrypts the message, processes the information, retrieves data and converts it into the message form.
10. Message is sent to the user using SMS gateway.
11. Message is then decrypted at the user side and responds accordingly.

### 4. TECHNOLOGY STACK

#### 4.1 Android Studio & java

Android Studio is an IDE which provides a development environment for android applications. Java has been used for developing the server side.

#### 4.2 Servlet API:

The Servlet API provides interfaces and classes that are required to build servlets. A **servlet** is a class that is used to extend the capabilities of servers that host applications accessed by means of a request-response programming model. Although servlets can respond to any type of request, they are commonly used to extend the applications hosted by web servers.

#### 4.3 SMS gateway

An SMS gateway is a Web site that allows users to send SMS messages from a Web browser to people within the cell served by that gateway. An SMS gateway can also serve as an international gateway for users with roaming capability, allowing SMS communication away from the home network.

#### 4.4 SQL Database

SQL stands for Structured Query Language. SQL lets you access and manipulate databases. SQL became a standard of the American National Standards Institute (ANSI) in 1986, and of the International Organization for Standardization (ISO) in 1987.

#### 4.5 Apache Tomcat Server

The Apache Tomcat software is an open source implementation of the Java Servlet, JavaServer Pages, Java Expression Language and Java WebSocket technologies. The Java Servlet, JavaServer Pages, Java Expression Language and Java WebSocket specifications are developed under the Java Community Process. The Apache Tomcat software is developed in an open and participatory environment. The Apache Tomcat project is intended to be a collaboration of the best-of-breed developers from around the world. Apache Tomcat software powers numerous large-scale, mission-critical web applications across a diverse range of industries and organizations.

### 5. Conclusion

E-Governance is the need of the hour and through this project we have tried to make it possible that our system can be used where, there is lack of internet connectivity and people can access all the services of government either online or offline.

It's future scope:

- Biometrics can be integrated.
- New module can be easily added.
- Aadhar number can be linked for authentication.

## 6. REFERENCES

- [1] Android development < <https://stackoverflow.com>>
- [2] Android Syntax <<https://developer.android.com/reference>>
- [3] Sawal Tandon, "Application of Incoming SMS to a Website, Control the Website to Send Bulk SMS", International Journal of Computer Science and Mobile Computing – 2012
- [4] Peersman, G., Cvetkovic, S., "The Global System for mobile Communications Short Message Service", IEEE Personal Communications, June 2000, pgs 15-23.
- [5] SMS Gateway <http://www.ozeki.hu/index.php>
- [6] JAVA <<https://www.w3schools.com/>>
- [7] Server <<http://tomcat.apache.org/>>