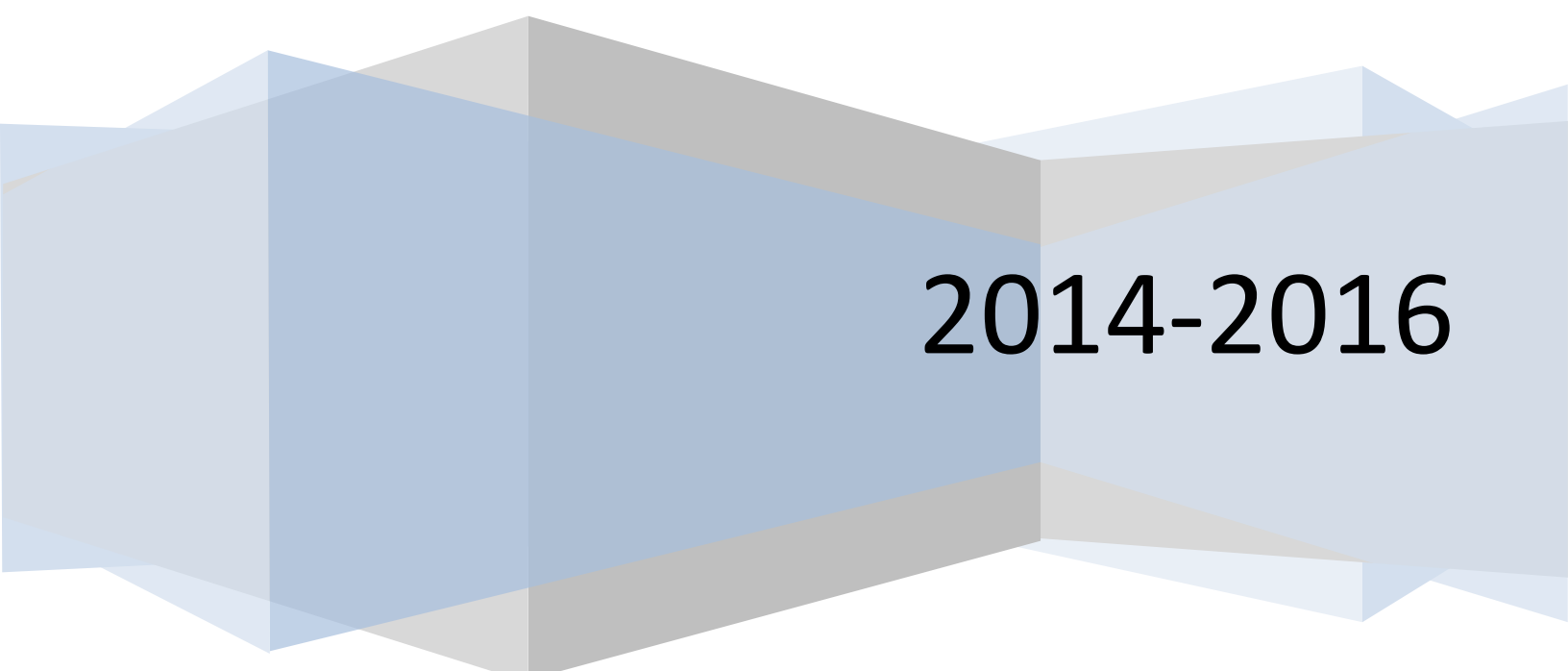


PERCCOM

# The Citizen App

An Application for Citizen-Participation

Md. Anowarul Abedin, Adebayo Adekola,  
Rajeshwari Chatterjee, Sumeet Thombre



2014-2016

# Table of Contents

---

## Contents

**Acknowledgement**

**Introduction**

MOTIVATION

SPIRIT OF CODE CAMP AND GREEN-IT:

**ARCHITECTURE DESCRIPTION**

**TECHNOLOGY**

**SUMMARY**

CUSTOMER FOCUS :

MOTIVATION FOR PARTICIPATION :

# Acknowledgement

---

We take this opportunity to express our gratitude to everyone who supported us during this project and without whose help and support this would not have been possible.

We would like to express our warmest thanks to Antti Knutas for his constant support and valuable advice during the code camp and for providing all the facilities necessary for the project.

We would also like to thank all the members who had participated in the code camp and were incredibly helpful along the way.

# Introduction

The **Citizen App** is a platform where citizens or “Netizens” can report incidents related to safety and facilities in their localities. While our aim is to make the application available both as a website platform and in the form of a mobile application, currently we are focused on developing the web-based application for a superior user experience.

The Citizen App provides an immediate option for reporting issued related to safety threats and availability of civil facilities such as (electricity, water, transportation etc.) in the locality. People can report incidents of crime, noise, traffic violations, etc. in real time. The data collected from the reports from the will be open for all, i.e. to say our application will work as an Open Data source for individuals or organizations (including government). We are also offering an API for third party integration of our application.

In addition, the statistics collected from citizens would be forwarded to law-enforcement agencies and civic authorities for assessment and future plan of action for better safety regulations and improved civic facilities for improving the level of good governance.

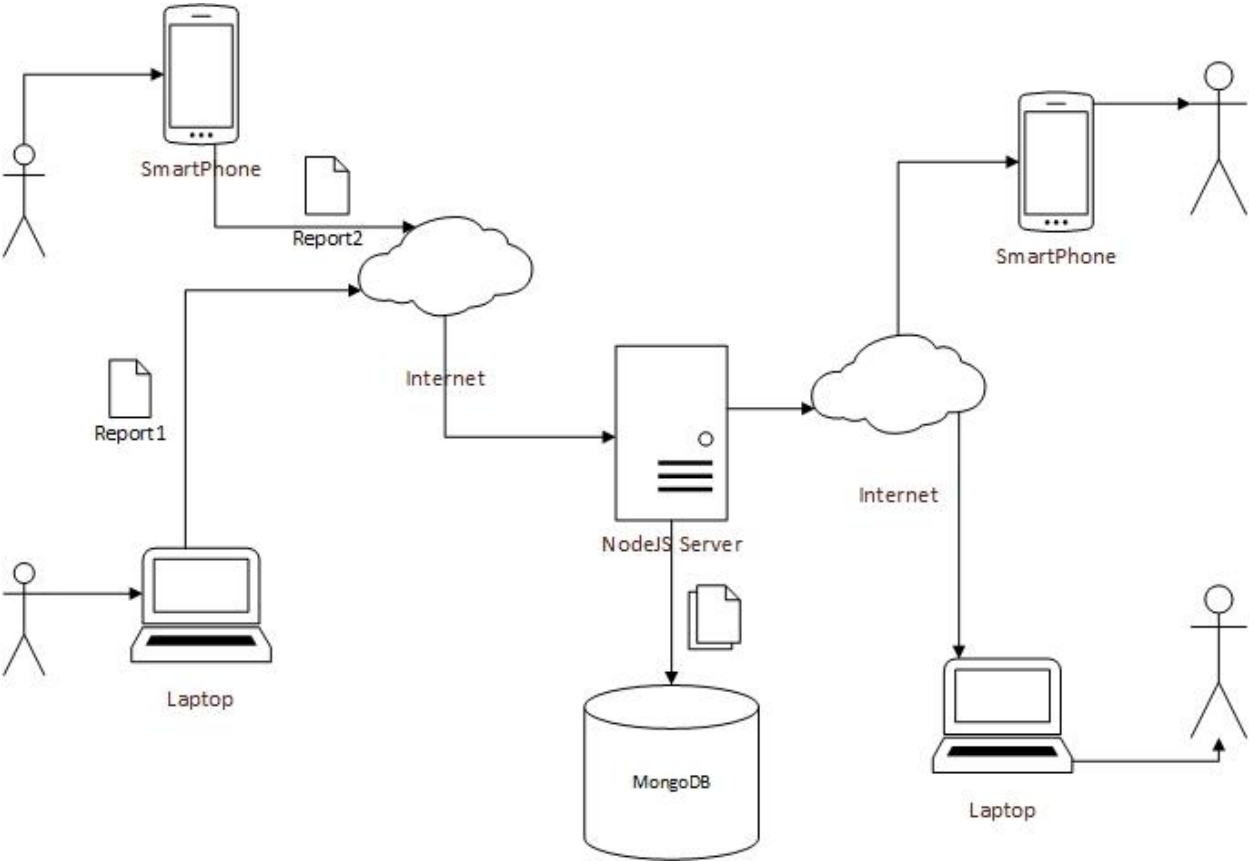


Fig.1: Application Flow

## MOTIVATION

The Application aims to provide a platform for the participation of citizens from across all section of society to voice their opinions continuously, while ensuring privacy and anonymity of the participants are maintained at all times. The application demonstrates the importance of public opinion and involvement of citizens and society collectively for improved governance and transparency.

## SPIRIT OF CODE CAMP AND GREEN-IT:

This main focus of the code-camp was on **Open Data, Citizen Science** and implementing **Green IT technology**. The Citizen App addresses all the three issues and therefore upholds the spirit of the code camp in all respect.

The application uses Open Data source such as Google Map, and it also exposes itself as an open data source allowing integration scope by third parties.

Citizen Science is the collection and analysis of data related to the natural world by members of the general public, typically as part of a collaborative project. Crowd-sourcing is one of the primary implementations of Citizen Science. The Citizen App adopts the idea of crowd-sourcing for data collection, analysis and making it in turn available and open for future use and analysis by other organizations. Organizations, NGO's, individuals etc could make use of the data for further future analysis for reporting, statistics, trend calculation and display.

Although, this application is not a direct implementation of **Green IT**, we have implemented several green coding techniques and thus worked towards achieving "Green IT". The Citizen App utilizes **Cloud Computing** for efficient resource utilization. Our application has more to do with "IT for Green".

As a collaboration platform which utilizes crowd-sourcing the application is a propitious interface for Knowledge Management. Current real-life scenarios for collection of information collection related to living standards and conditions are involves a lot of human resource . Our platform would provide a seamless application for collection and integration of information remotely thus requiring minimal human-effort thus reducing Carbon-Footprint.

Summarizing the various features of the Citizen App:-

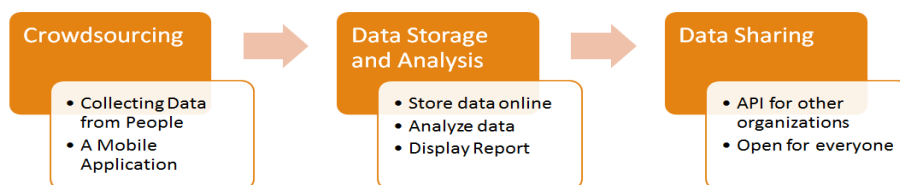
- The Citizen App could be used to ensure environmental cleanliness, promote citizen participation including their opinion for availability of better civic amenities.
- The Citizen App could be a useful platform for gathering statistical information and citizen's opinion in a more resource efficient way.
- It is a platform for making an effort towards improving the quality of life of citizens

## Architecture Description

---

The Citizen App offers its services through 3 major processes:

- Crowd-Sourcing
- Storing Data, analyzing and Displaying Report
- Sharing the data for future use



**Fig 2:** Application Processes



**Fig. 3** Parameters for Reporting

### FEATURES:

- Incident Reporting
- Scope for providing additional commenting for better insight.
- Viewing a detailed and comprehensive Report for the location
- Poll of the Week based on a current issue.

- Geo-location features which improves usability quotient.
- Statistical Report based on people's feedback.

S/N	Name	Category	Implemented	Comments
1.	Submit report at current location	Web App	YES	
2.	Submit report on different location	Web App	YES	
3.	View reports on current location	Web App	YES	
4.	View reports at other locations	Web App	YES	buggy
5.	View Reports by category	Web App	NO	
6.	View reports by distance	Web App	YES	
7.	View Reports by type	Web App	NO	
8.	Get Report by ID	API	YES	
9.	Get Reports by Type	API	YES	
10.	Get Reports by Category	API	YES	
11.	Get Reports by Location & Distance	API	NO	

Table 1: Features of the CitizenApp by type and implementation

## Implementation

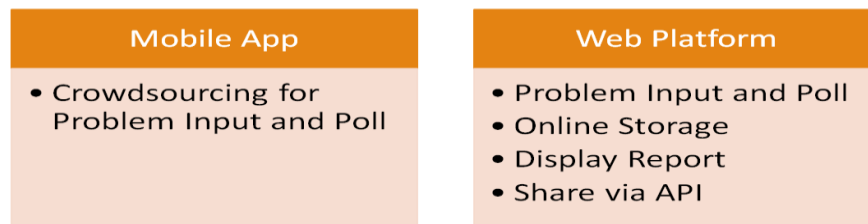


Fig. 4: Mobile and Web-Based Platform Implementation

We had divided our implementation in two phases –

### DESIGN PHASE-

In the Design phase we had considerable discussions among all the team members for a clear idea of the **requirement analysis**. Once we had the first UML in place we spent our time in understand the **architecture** for our project and also get a clear idea of the **implementation plan** and analysis of the **priority features** which are absolutely essential for the success of the project implementation. We had also charted out a Test plan for bug-fixing and updates. Based on our analysis of the performance necessary for the application and the framework to be implemented we had agreed upon to implement the MEAN stack for our development purpose which has the following features:

- **M-** Mongo as the Database
- **E-** Express.js as the Web-Framework
- **A-**Angular JS as the Front-End Framework.
- **N-**Node.js as the server Framework.

### DEVELOPMENT PHASE-

Once we had started with the actual coding phase we agreed to implement the **Agile Programming style** for execution of coding in sprints and we have stuck to MVC pattern for the implementation.

For our development, we have primarily used the Cloud 9 IDE platform, which has excellent features for collaborating for peer-programming apart from an easy environment setup.

We have used the **Bootstrap Framework** for the front-end, which greatly helped in reducing the time and effort for setting up the web-page. We have implemented our front-end in Angular JS which has excellent feature for REST implementation, Data-Binding and Dependency Injection and its ability to extend HTML all of which makes it an excellent choice for quick front-end development.

We have used several plug-in for our application such as the **Google Map Geo-Location API** for retrieving the location statistics of our user. Apart from these we have also made use of the **Angular JS Core module** and **Angular JS route provider** module for routing and deep-linking services

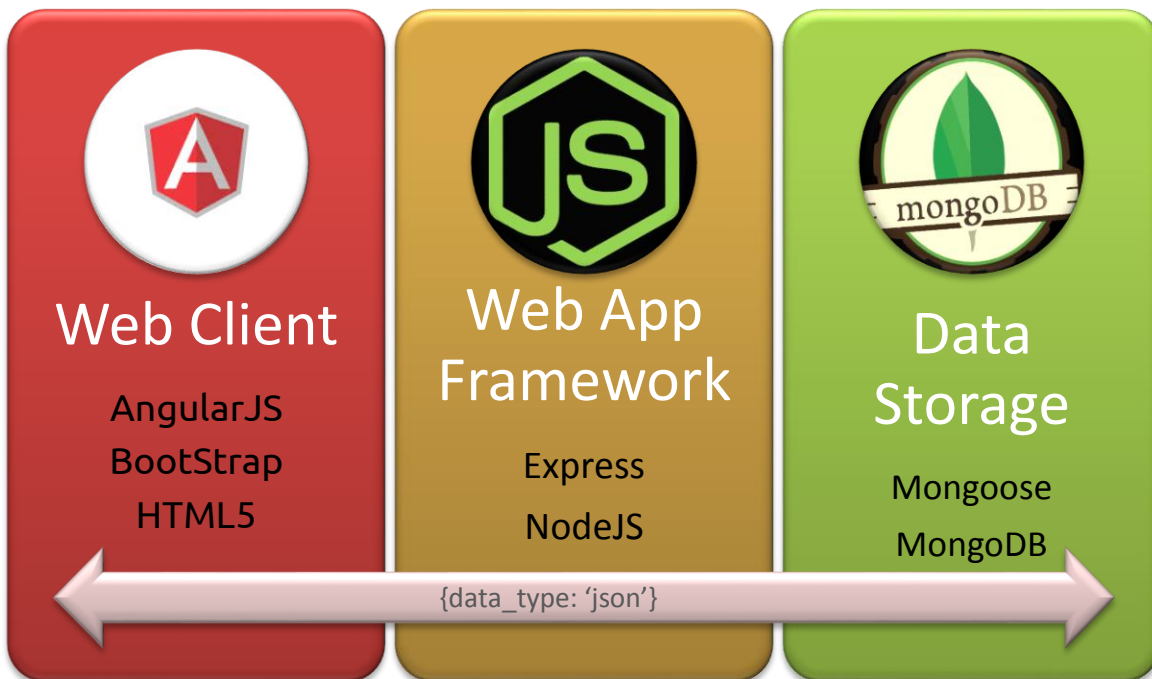


For the backend we have used MongoDB for our Database and the MongoDB Object Modeling Tool-**Mongoose** for schema based solution for application modeling the data.

**Android Development**-We have used the Android Studio IDE, with built in features of Android SDK tools and Android 5.0 emulator with Google API's and the initial phase testing was done using the Emulator. Currently, the full version of the application is not available for the mobile platform.

## Technology Stack

---



Given that the code-camp is centered around the use of the MEAN stack, it was only natural that we use it to implement the platform. Besides, as shown in the figure above, the fact that the entire stack is based on javascript makes it a breeze to work across the different levels of the stack. JSON (JavaScript Object Notation) was used as the data format from the front-end up to the MongoDB data store.

The web client was declaratively defined using HTML5 which allows us to take advantage of the latest specification of the markup language. We took advantage of the Geo-location API which comes with HTML5 to detect the location from the user access the web application.

Since HTML5 declaratively handles the structure of the web pages, Bootstrap was used to style the views to make them attractive. Bootstrap

makes it quite a breeze to decently style HTML5 templates and make them user-friendly.

At the core of the front-end's functionality is the AngularJS framework with which we built the CitizenApp as a SPA (Single Page Application). Being a SPA means that all the views in CitizenApp are on a single page and the user experience is fluid since pages are not always loaded when the views have to be switched.

ExpressJS as a web framework was used to implement the routes for RESTful endpoints of the API to which the Front-End also connects. Using express, we implemented the routes (GET and POST) as well as Parsing the body of requests and logging requests when they hit the web server.

Mongoose and MongoDB making up the base of the stack were used for data persistence. MongoDB being a document-oriented database stores JSON objects which is the data-exchange format within CitizenApp. Mongoose allowed us to define schemas for the database so that all data going into the database conforms to the same structure.

## SUMMARY

---

### CUSTOMER FOCUS:

Our primary focus in this code camp has to develop a social-app in order to bridge the divide between governance and the citizens and provide a platform for everyone to voice their grievances.

As a first initiative we have broadly categorized our target group as below –

- People living in a locality who would like to express their grievances with regards to safety regulations and civic amenities.
- Law –Enforcement agencies and Civic Authorities who could use the data available for informed decision making
- People moving in to a different locality could be pre-informed on the living conditions and safety measures of their new neighborhood
- Third-party organizations/NGO etc. who might want to make use of the data available from our application for research, analysis or future development.

Finally the code camp was a great learning experience for all the team-members and we had an opportunity to experiment with new technology and pick up essential coding skills and techniques. It was also a good way to learn how the development work actually takes place in the real-world scenario.