

# Real-time Grievances Resolution System

Chithra Devi<sup>1</sup>, Vedhanayaki<sup>2</sup>, Valli<sup>3</sup>, Naganandhini<sup>4</sup>

Department of CSE, NSCET, pca90chithrapraveen@gmail.com

Department of CSE, NSCET, vedhakrish1999@gmail.com

Department of CSE, NSCET, viswanathanvalli27@gmail.com

Department of CSE, NSCET, nandhini27nallamuthu@gmail.com

**Abstract**—Building a real time grievances resolution system to overcome the delays in fulfilling grievances to the people. Petition filed to Government offices, generally takes a long time before they are resolved. It is mainly due to large number of petitions that are collected manually. Implementing a digital assistance through, in-app grievance submission and tracking could help address this problem and could also give the people, a transparent and reliable source of communication with the local administrations.

**Keywords**—Flutter, MongoDB, Node JS

---

## I. INTRODUCTION

To overcome these problems, now we made an online petition system. By this online petition application, we can easily submit petition to the higher officials. We can able to monitor our petition status and made report if they are not take any actions for our petition. By this way, we can able to solve our issues quickly. Making online petitions has become a ridiculously easy process that anyone can do at the click of a button. Online petitions may have a small expected direct impact, especially when the cause is important and the decision-makers receive the messages.

In the average case, I think a good rule of thumb is that petitions aren't the best use of activist time unless you do them for leisure or self-education. The bigger effects of petitions may be indirect way of raising awareness among those signing. These impacts may be decently worthwhile for some groups. Creating petitions for purposes of awareness raising probably wouldn't work well. I wouldn't discourage petition-signing, but I would encourage more ambitious Project's to those able to undertake them. Previous research has discovered some challenges and issues of developing online government application which involved citizen participation.

The absence of user involvement in the design and development online government usually become a barrier. E-petition represents as a key component, as it is the way to adapt government decisions to the real expectations of citizen. To solve these problems we designed an application. Through this online petition application, we submit our issues and also see the status of our petition.

## II. MAIN PROBLEM

Based on the description above, the problem is how to design and build an online petition application for our society. It should be in real time. It need government permission to connect the higher officials in this application. Then we can able to submit our issues to the government.

## III. OBJECTIVE

An objective then can be defined from the main problem above that is build a dedicated real-time multi-platform online petition application using flutter which is used by all peoples. Through this we can able to convey our issues to the higher official's directly.

## IV. SCOPE

The scope of system should be declared before move advancing to the next step. System scope are as follows:

1. Design and construction of this online petition application.
2. This system was developed by using Flutter, Node JS.

3. Database of this online petition application using MongoDB.
4. Used by less than 100 people (for testing purpose)

### **V.SYSTEM DESIGN**

The system will be built in two sides; server and client side. Server side has service and middleware, while the client side we have application. Mobile app development is the act or process by which a mobile app is developed for mobile devices, such as personal digital assistants, enterprise digital assistants or mobile phones.

Client will communicate to server through Node JS which act as a middleware of server. Node.js is an open-source, cross-platform, back-end JavaScript runtime environment that runs on the V8 engine and executes JavaScript code outside a web browser. Node JS is a platform built on Chrome's JavaScript runtime for easily building fast and scalable network applications. Node JS uses an event-driven, non-blocking I/O model that makes it lightweight and efficient, perfect for data-intensive real-time applications that run across distributed devices.

Server's services are using the Node.js with MongoDB. Development teams of all sizes use MongoDB because: The document data model is a powerful way to store and retrieve data that allows developers to move fast. MongoDB's horizontal, scale-out architecture can support huge volumes of both data and traffic.

#### **A. Block Diagram**

Here is a block diagram of server – client communication

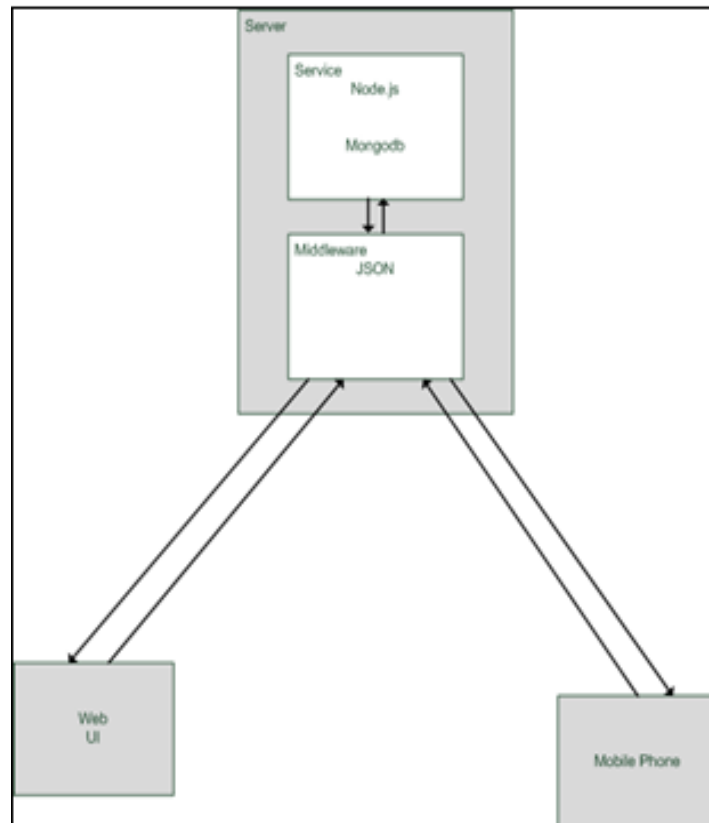


Fig. 1 Server – client block diagram

**B. Flow Chart**

Here is a flow of chat overallprocess.

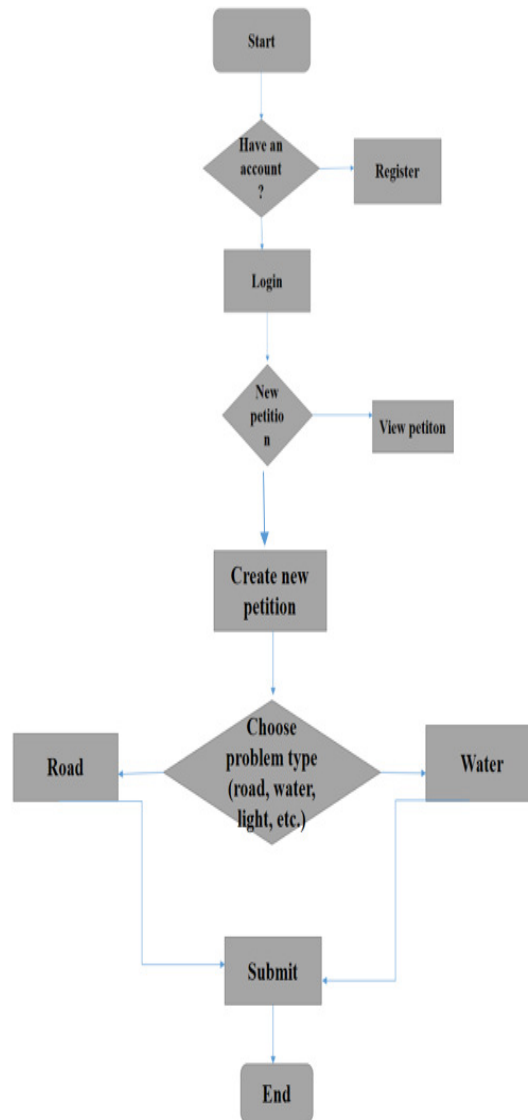


Fig. 2 User interface design for online petition application

The explanation for each flow from picture above is as follow:

1. When the main page is accessed by user it will show the login page first, instead of show the main page directly. User login credentials is required. If the user doesn't have an account yet, then they must register first.
2. After login, we may view all the petitions we have submitted and also we can able to view other public petitions too.
3. Create new petition by clicking new petition button, then we have to give problem title, description, problem type, problem/issue related photos, etc.
4. Then we submit the petition, it goes to the corresponding higher officials.
5. After submitted the petitions, we can able to view the status of our petition.
6. Through this we can make our work easily and quickly.

C. **Data Flow Diagram**

Here is a data flow diagram to describe overall process of chat application data.

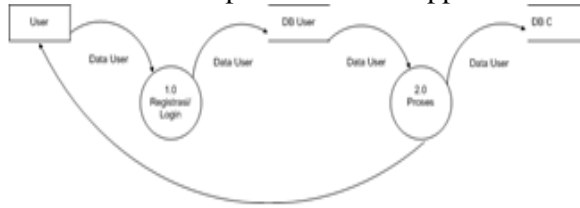


Fig. 3 Online petition application data flow diagram

D. **Database Table Data Structure**

Here is database table data structure for each table user and chatroom.

**TABLE I**

**USER DATA STRUCTURE**

No	Field Name	Data Type	Explanation
1	_id	Object Id	User Id
2	Mobile number	Integer	Cell number
3	OTP	Integer	One Time Password
4	Username	String	Username

[1]

**TABLE II**

**PETITION DATA STRUCTURE**

No	Field Name	Data type	Explanation
1	_id	Object Id	Application Id
2	applicationTitle	String	Petition ID
3	applicationDescription	String	Petition description
4	applicationState	String	Application/Petition status
5	Applicant	String	Username
6	applicantID	Integer	User ID
7	Latitude	Integer	Latitude
8	Longitude	Integer	Longitude
9	photoOne	Image	Image we inserted
10	photoTwo	Image	Image we inserted
11	applicationType	String	Petition type
12	applicationSubType	String	Petition sub type

## VI.IMPLEMENTATION

Online petition application, overview of our petition application.

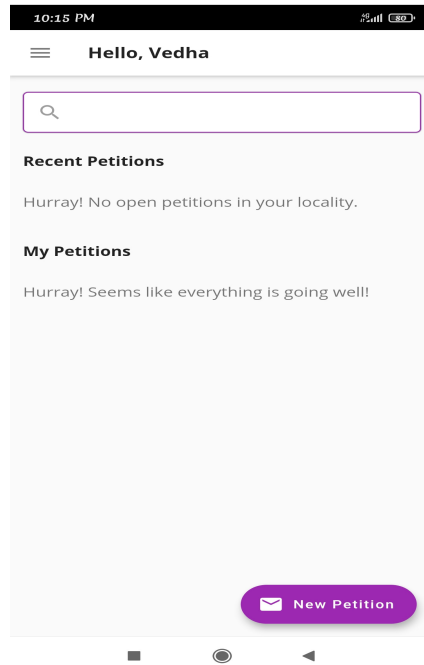


Fig. 4 Online petition application

The figure shown above, first we don't have any petition. After submitting petition we can able to view our petition and also view other public petitions.

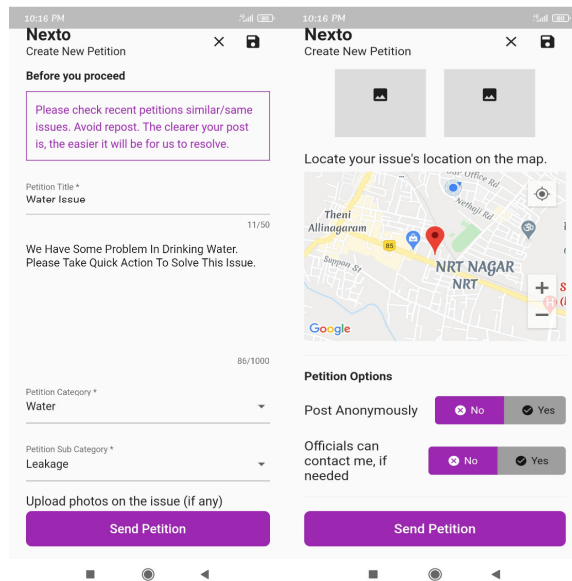


Fig. 5 Create and submit petition

The figure shown above, this is online petition form. Here we submit our issues. Through this we can able to convey our problem to the higher officials.

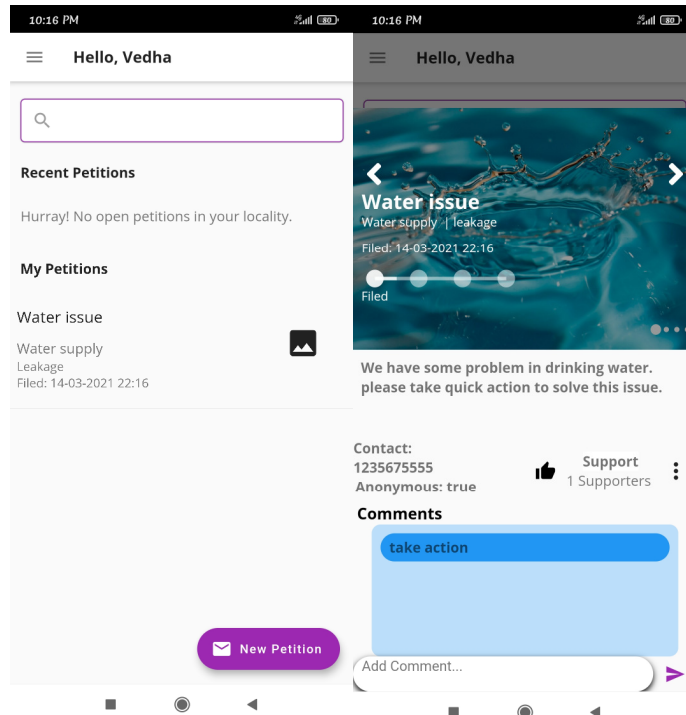


Fig. 6 View all petition

Fig. 7 View particular petition

The figure shown above, here we can able to view our petition. We also able to view public petitions. In this, we have number of supporters, comments, etc. to convey our support to that particular petition.

## VII. TEST RESULT

As the result, the main features of these online petition application sends and receives the petitions very well. Online petition application built with Node.js, MongoDB and Flutter running much faster in order to achieve real-time applications compared with the application built by using java or kotlin. Online petition application going to play a vital role in future. It must solve everyone problems and no one can miss use their power. Here we using flutter for efficient use of users. By using flutter, we can able to develop applications for both android and ios.

## VIII. CONCLUSION

Here we build an online petition application for the welfare of our nation. By using this we can able to convey our problems/issues to the government. In existing system, they have some problems in user convenience and also in user designs. Due to that petition users rate get reduced. To increase user rate we gave solution in our application. In our application we included status viewer. Using this we can able to view our live status of our petition. By this we solve the problems in existing system.

## REFERENCES

- [1] Hibah PITTA, "User-Centered Design in Emerging Citizens Initiatives: A Study of E-Petition in Indonesia" (Publikasi Terindeks Internasional Untuk Tugas Akhir Mahasiswa) 2018 funded by DRPM Universitas Indonesia No.1871/UN2.R3.1/HKP.05.00/2018
- [2] A. Macintosh, "Characterizing E-Participation in Policy-Making," in Hawaii International Conference on System Sciences, 2004.
- [3] T. Dargan and F. Evequoz, "Designing engaging e-Government services by combining user-centered design and gamification: A usecase," in Proceedings of the European Conference on e-Government, ECEG, 2015, pp. 70–78.

- [4] [M. K. Joachim Åström, HilleHinsberg, Magnus E. Jonsson, “Crisis, Innovation and e-Participation: Towards a Framework for Comparative Research,” in 5th IFIP WG 8.5 International Conference, 2013, pp. 26–36.
- [5] [E. Eshet and H. Bouwman, “Context of Use: The Final Frontier in the Practice of User-Centered Design?,” *Interact. Comput.*, vol. 29, no. 3, pp. 368–390, 2017.
- [6] [M. Maguire, “Methods to support human-centred design,” *Int. J.Hum. Comput.Stud.*, vol. 55, no. 4, pp. 587–634, 2001.
- [7] L. Heerma van Voss, “Petitions in social history,” *International review of social history. Supplement*; 9,p. 234 p., 2002.
- [8] Y. Zheng, “Explaining Citizens’ E-Participation Usage: Functionality of E-Participation Applications,” *Adm. Soc.*, vol. 49, no. 3, pp. 423–442, 2017.