

# MVVM ARCHITECTURE BASED SOCIETY MANAGEMENT SYSTEM FOR ANDROID DEVICES

Vikas Pandey <sup>1</sup>, Swapnil Kadlag <sup>2</sup>, Jaideep Dave <sup>3</sup>, Dr. Vinayak Shinde <sup>4</sup>

*Student, Department of Computer Engineering Shree L R Tiwari College of Engineering Mumbai University Mumbai, India <sup>1,2,3</sup>*

*Professor, Department of Computer Engineering Shree L R Tiwari College of Engineering Mumbai University Mumbai, India<sup>4</sup>*

*Email: vikas.pandey9969@gmail.com, swapnil.kadlag.27@gmail.com, jaideepd3@gmail.com, vdshinde@gmail.com*

\*\*\*\*\*

**Abstract:** - Housing Society Management System aims to make the current situation in society simple and efficient. Society management apps go a long way in ensuring the smooth functioning of a society and improving its efficiency while reducing the committee's workload and manpower during a pandemic such as the current covid-19, it's not appropriate to step out of the house be it for the residents or the managing committee members for tasks such as viewing the notice board or conveying an important message to residents. housing society management system eases all the problems within the society by providing a variety of facilities such as daily notices, monthly meetings alerts, important event messages via push notifications. whereas it also gives access to the residents to register their suggestions and complaints. And also view the rules and regulations which are set by the housing society management committee. this application is based on the MVVM architecture model. The MVVM (Model-View-View Model) pattern helps to completely separate the business and presentation logic from the UI, and the business logic and UI can be clearly separated for easier testing and easier maintenance. Enhance the reusability of code. All modules are independent which improves the testability of each layer. Makes project files maintainable and easy to make changes.

**Keywords:** *Society Management, Residents, Push Notifications, Rules, Monthly Meetings.*

\*\*\*\*\*

## I INTRODUCTION

A co-operative society is a voluntary association of individuals having common needs who join hands for the achievement of common economic interest. Its aim is to serve the interest of the poorer sections of society through the principle of self-help and mutual help. The main objective is to provide support to the members. Nobody joins a cooperative society to earn profit. People come forward as a group, pool their individual resources, utilize them in the best possible manner, and derive some common benefit out of it [1]. 'Manual' managing activities has become outdated in the digital age. Technology has taken over every aspect of an individual's life, whether it is in the domestic sphere, at the workplace, or within the community. A perfect illustration of this is the management of residential societies/complexes with the aid of a relevant Society Management System. This is because the modern mindset demands more features, better administration, in order to ensure that life is hassle-free. people having a very busy schedule towards their work, everyone having different working hours and holidays so they are unaware of what is going on in their societies like events, meetings etc. this problem is solved by using this android mobile application. This project mainly uses "Push Notification Technology"

through Android platform [2]. Our system overcomes the drawbacks of the traditional system by forming a network of people online where things can be done on a click thus reducing the dependency on other people and bringing translucency in the management system of the society.

## II MOTIVATION:

Develop an android application which acts as a connecting bridge between the Management committee members and the Society Residents. To make use of push notification technology to maximize the reach with the society members. develop and implement maintenance or expense reports in online manner so as to it can be sent to the residents in timely manner. Implement rules and regulation for the society members which can be accessed easily. implement complaints and suggestions so that the residents of the society can provide to the managing committee members in online manner and from one single app. implement Polling within the application which helps the managing committee members to get the fair idea as in to proceed or not with the proposed idea. To make an application to match the needs of the tech savvy generation. For maintenance of the society with fast response time and resolve the issue in efficient manner.

### III LITERATURE SURVEY:

Android application which contains features like Notice Board, Calendar, Contact, Maintenance, Financial Report, Complaint or Suggestion. Helps to overcome the drawbacks of the existing system, the application provided an efficient way. A special feature called “Push through Notification” which helped in efficient usage of roles to the society members [3].

Web application helps in improving the sociality of the society by helping them interact with each other by becoming a member of a cultural or sports group. Due to this, the residents are always updated of the current affairs regarding the society matters and they can also participate in various events. It also helped the residents to give complaints and suggestions via the application instead of personally meeting the chairman. Also once the complaint or suggestion is resolved, it is updated in the application [4].

Due to the various technologies and advancement in it android application contains features like QR code authentication, Online Payment, Society Maintenance Amount. To surmount the drawbacks of the subsisting system, they provided a more astute and efficient way to handle the critical issues by truncating efforts and advancements in reliable communication. Different functions in the society like Complaints, Notices, Meetings, Rules, Suggestions, and Miscellaneous Contacts were available within a single visual perception so that users can visually examine it and make utilization of it whenever is obligatory [5].

To overcome the drawbacks of the current system this project provided a simple and efficient way of communication via an android app. This application had feature which enabled the person to person communication. Different functions within society like Complaints, Notice, Meetings, Rules, Suggestions, and Miscellaneous Contacts were available at a glance so that the residents can use any of the features easily. This application also reduced the work of the admin by providing calendar events so that admin did not have to keep track of the information's reachability [6].

Various features like auditing, parking, alert button, notice board are some of the features in this android application it also contains the user module & Admin module. Admin maintained a maintenance list of all the flats in a society. The user will get a prompt on UI about their pending maintenance payment & user has to pay predefined maintenance amounts. Admin manually clicks on respective name flats and sends the update. The parking UI displayed a detailed chart of the parking slot status to the user. So, accordingly, they can book parking slots [7].

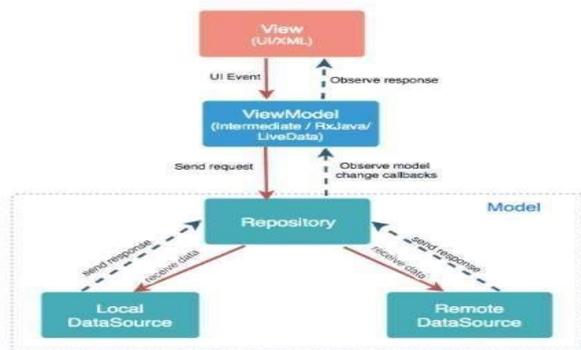
### IV EXISTING SYSTEM:

In current situations, housing society management authorities use a traditional way of communication which include a common notice board system operated by secretary of the housing society. Other problem in existing society management system is that suppose if there is dispute among society members and to sort out the issue between them rules and regulations of the society needs to be available with society members so that they can refer it instantly and sort out the issue, but in current scenario to access the rules and regulation of the society person needs to go the management committee member to collect the latest rules and regulations of society, which leads to more frustration and is time consuming. whereas whenever there is need for the managing committee to send any update or message which needs to be passed on to the society members a circular need to be placed at noticeboard but there is high chance that all the persons living in the society might not get the update on time as they might not have read the circular on the notice board. Many a time due to busy schedules and the fast life of the city society members tend to forget to pay the society bills and so the bill payment of each member needs to be notified by the treasurer of the society on timely basis so that the resident does not default on the payment of the society maintenance bill or different other bills. residents of the society more often or not face various issues in society and for that they need to contact the managing committee to get their problems heard and get resolved but in the traditional working way residents need to keep waiting for the days at stretch to get update and for that they need to approach secretary, chairman or other management committee member.

### V PROPOSED METHODOLOGY:

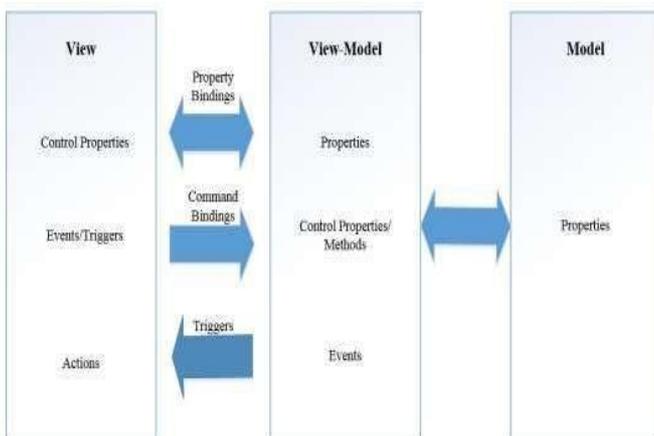
This application is based on the Model–view–view model architecture (MVVM) is a software architectural pattern that facilitates the separation of the development of the graphical user interface (the view) – be it via a markup language or GUI code – from the development of the business logic or back-end logic (the model) so that the view is not dependent on any specific model platform. The view model of MVVM is a value converter, meaning the view model is responsible for exposing (converting) the data objects from the model in such a way that objects are easily managed and presented. It handles most if not all of the view's display logic. The view model may implement a mediator pattern, organizing access to the backend logic around the set of use cases supported by the view. MVVM architecture is a Model-View-View Model architecture that removes the tight coupling between each component. Most importantly, in this architecture, the children don't have the direct reference to the parent, they only have the reference by observables.

As shown in the **figure 1** the MVVM architecture consists of 3 main modules they are as follows:



**Figure 1: MVVM architecture**

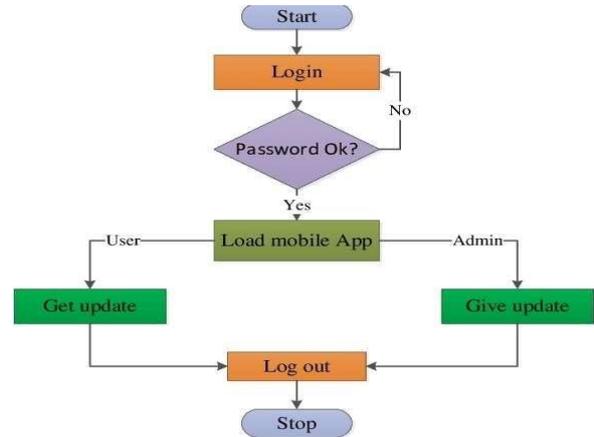
**Model:** It represents the data and the business logic of the Android Application. It consists of the business logic - local and remote data source, model classes, repository. **View:** It consists of the UI Code (Activity, Fragment), XML. It sends the user action to the View Model but does not get the response back directly. To get the response, it has to subscribe. **View Model:** It is a bridge between the View and Model (business logic). It does not have any clue which View has to use it as it does not have a direct reference to the View [8].



**Figure 2: Data binding in MVVM**

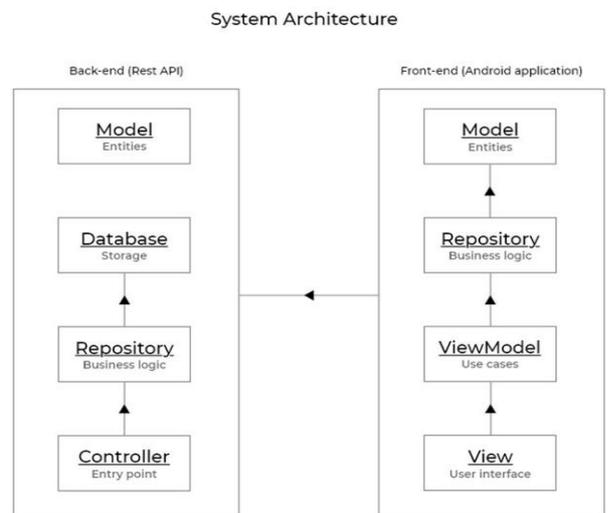
The figure 2 represents how the databinding takes place in the MVVM model. Databinding is a new mechanism introduced in MVVM. It allows the view directly bound to the properties and operations of the View-Model. With the databinding, View-Model component does not have to notify the view changes via code and the view knows data is loaded and shows the data by view itself. For example, In MVP and MVC architecture, after the data loaded, the presenter and controller will set the view in code. With databinding mechanism, the view is bound to this data and when data is loaded, view will change automatically. The databinding between the View and View-Model can be directional and

bidirectional. When the data bound in view is changed, the data in View-Model component also know the change [9].



**Figure 3: System Flow Diagram**

The figure 3 shows the system flow diagram as in how the user once login to the application and the process behind it. User will login to the system on the basis of its position in the society, such as a managing committee member will login in the Admins section and the normal resident will login as the user. Admins or the managing committee members will provide the updates to the residents and the residents will receive all the updates from the managing committee members.



**Figure 4: System Architecture**

As shown in the figure 4 it refers to the system architecture diagram of the application where in the back end and the front end are shown along with the submodules.

**Back-End (Rest API):** Back end of this project will be based upon MVC Architecture, The Model-View-Controller (MVC) is an architectural pattern that separates an application into three main logical components: the model, the view, and the controller. Each of these components are built to handle specific development aspects of an

application. MVC is one of the most frequently used industry-standard web development framework to create scalable and extensible projects.

**Model:** The Model component corresponds to all the data-related logic that the user works with. This can represent either the data that is being transferred between the View and Controller components or any other business logic-related data. For example, a Customer object will retrieve the customer information from the database, manipulate it and update it data back to the database or use it to render data.

**Database:** This relates to the database & storage related components which are used for back-end of our application

**Repository:** This is a well-documented way of working with a data source. A repository performs the tasks of an intermediary between the domain model layers and data mapping, acting in a similar way to a set of domain objects in memory.

**Controller:** Controllers act as an interface between Model and View components to process all the business logic and incoming requests, manipulate data using the Model component and interact with the Views to render the final output. For example, the Customer controller will handle all the interactions and inputs from the Customer View and update the database using the Customer Model. The same controller will be used to view the Customer data.

**Front-end (Android application):**

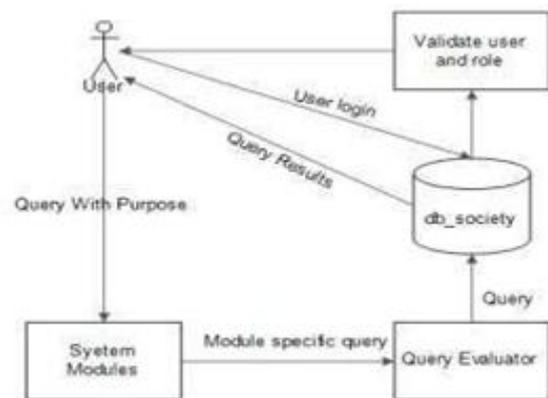
Model-View-View Model (i.e. MVVM) is a template of a client application architecture, proposed by John Gossman as an alternative to MVC and MVP patterns when using Data Binding technology. Its concept is to separate data presentation logic from business logic by moving it into particular class for a clear distinction. **Model:** is the logic associated with the application data. In other words, it is POJO, API processing classes, a database, and so on. **View:** is actually a layout of the screen, which houses all the widgets for displaying information. **View Model:** is an object which describes the behavior of View logic depending on the result of Model work. You can call it a behavior model of View. It can be a rich text formatting as well as a component visibility control logic or condition display, such as loading, error, blank screens, etc. Also, it describes the behavior that was initiated by the user(text input, button pressing, swipe, etc.). This application is implemented using IDE Android studio.

Android has a built in support for SQLite database implementation for the local storage of data on the device. Application is developed by following MVVM (Model-View-View model) architecture pattern for development. In android Model represents all java classes i.e. supporting classes in an application. View represents all the activities in application having the layouts defined using Xml. And View model represents all the backend running services APIs, Interfaces, Adapters. Software's used: Kotlin/Java ,SQL DB , Spring Boot Rest Framework, Xml, Android Studio , IntelliJ Idea, SQL server , Google Cloud Messaging.

Accordingly notices, meetings scheduled by admin, Notifications are generated and will be pushed to the resident's devices And complaints by residents will be sent directly to Admins. In order to achieve reachability to admin. Notifications are pushed so information reaches in reliable and transparent way.

### VI PROPOSED SYSTEM:

To address the shortcomings of existing systems this project serves a solution for the smart way of communication through an android application which helps in the faster decision making and maximize its reach to the residents of the society. The figure 5 Refers to the system behavior where in it validates the user login and provides the result accordingly.



### How System Works:

New member will register on the application by providing the details. If the user is already registered, then user can directly login into the system. The system will differentiate whether the user is 1 Admin 2 Secretary 3 Treasurer 4 Resident.

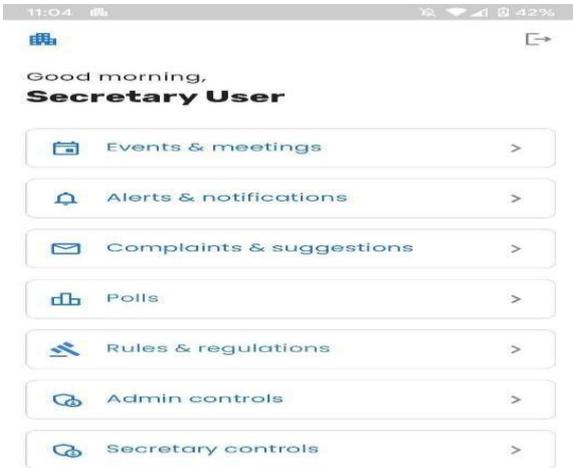
### Once the login is done based on the type of the user application dashboard will appear:

Administrator can insert posts, push notices, update the application, add or remove the users. generate username and password for all the member of this application. Secretary will be able to send the alerts and notices to the residents related to any events and meeting. can check the suggestion and complaints to work on it. modify the rules and regulation of the society within the app and display it. treasurer will be able to send the bills and notifications related to the financial aspect of the society to the user. And also will be able to access the suggestion and complaints which are raised by the residents. Resident will be able to view the daily updates and notices, view the rules and regulation of the society, raise complaints and provide the suggestions within the application.

### VII RESULTS & DISCUSSION:

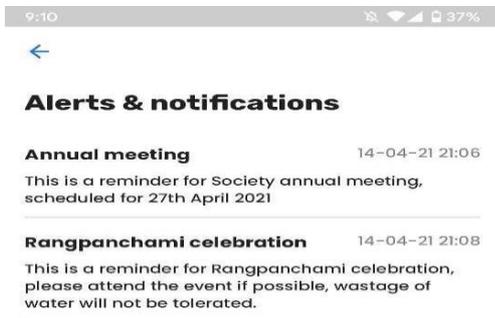
The snapshot refers to the secretary login as shown in Figure 6 wherein the secretary can do the various tasks such as can

create event and meeting, provide alerts and notifications. View complaints and suggestions so that it can be worked upon at the earliest.



**Figure 6: Secretary Login Screen:**

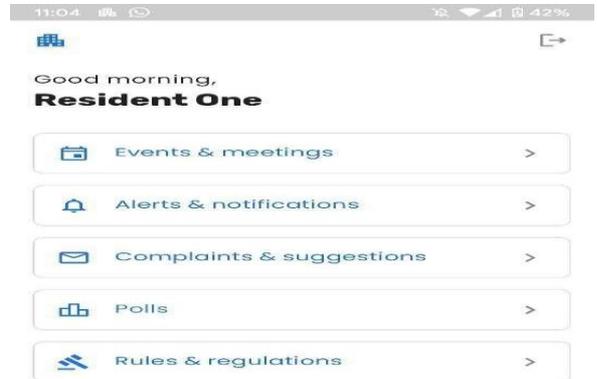
Multiple Alerts and notifications sent by the secretary as shown in **Figure 7** which helps secretary to keep track of the number of alerts and what all notifications have been sent previously.



**Fig 7: Alerts and notifications**

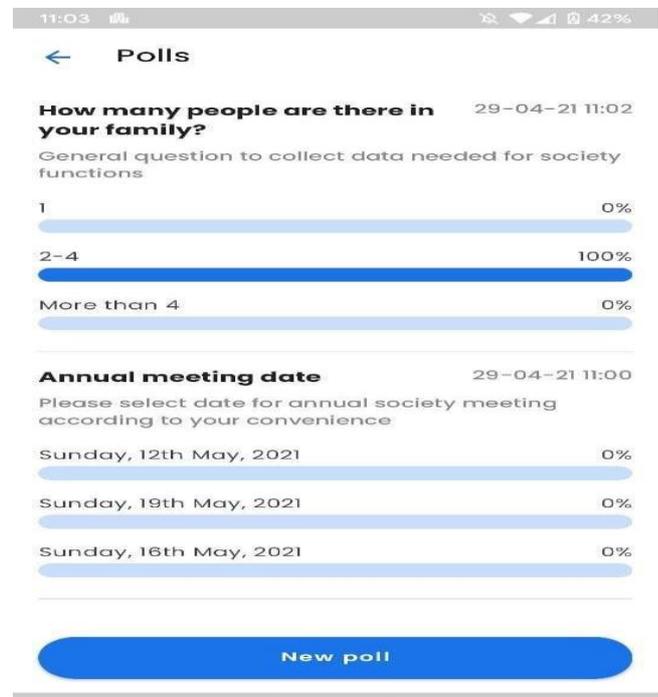
Refers to the Resident login as shown in figure 8 where in the resident Can see the rules and regulations, take part in polls and Give suggestions and complaints. And also view all the alerts and the notifications sent by the managing

committee. Since all the features are available in one single mobile application this helps members of the society in easy access to management and get the work done in fast and unified manner.



**Fig 8: Resident login Screen**

Polls can be scheduled or conducted for the society members by the secretary as shown in figure 9 which helps the managing committee members to get a fair idea regarding how many people are ready to accept the proposed solution or any other instance.



**Fig 9: Polls by Managing committee**

Rules and regulations of the society are made available within the app itself as shown in **figure 10** so as any person

within the society can refer to it.



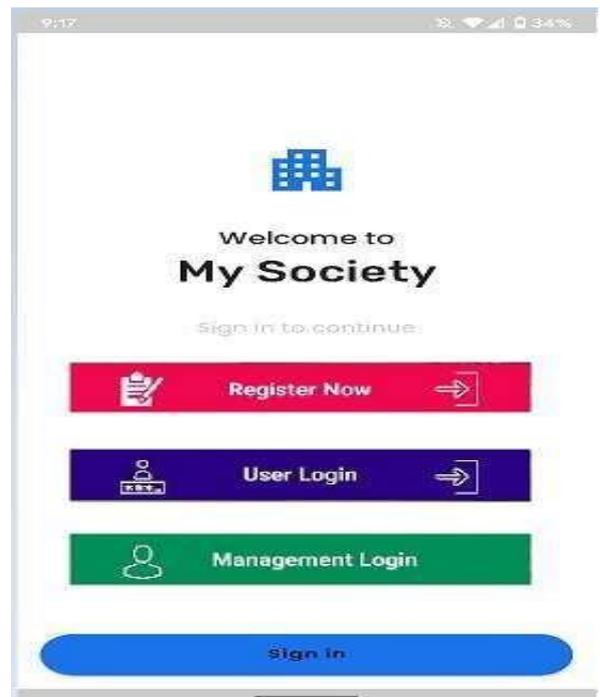
**Fig 10: Rules and Regulations**

The residents can also raise the complaint as shown in **figure 11** below. In case of any issue within the society.



**Fig 11: Complaints by the Residents**

Based on the various roles people can login as well as register on the application as shown in **figure 12**. It gives the person to register on the applications, user login is for the Residents of the society and the Management Login is for the Admins and the Managing committee members.



**Fig 12: Multiple Login Options (Main login Screen)**

**VIII CONCLUSION:**

The application provides a hassle free means of communication to conduct and regulate the day to day tasks in the society by the managing committee members be it the secretary, chairman and the treasurer and also it acts as the bridge between the society residents and the managing committee using affordable, easily available and customizable android technology. Since the android mobile is available with the majority of the people this mobile application fulfills the needs of every society in terms of by sending the notices, alerts via push notification technology, whereas the rules and the regulation of the society are at the fingertips of the resident so that it can be referred easily and also complaints and suggestions by the residents of the society can be raised and taken care by the management committee members within a short span of time. Hence Housing society mobile application is truly a need of the hour for the many societies. In this fast paced world now majority of the things are done via mobile applications. hence this application serves the purpose and the requirement of the present day needs.

**REFERENCES :**

[1] I. government, "Statistical Year Book India 2016," Indian government, New Delhi , 2016.  
 [2] M. Riadh, "Notification System to Students using an Android," International Journal of Computer Applications, vol. 140, pp. 22-27, 2016.  
 [3] S. Gavhane, R. Vatharkar, S. Sonar and P. Patil, "Study

of Implementation of Society Management System," International Journal of Computer Applications, pp. 34-36, 2015.

[4] S. Raut, P. Pawar, M. Shaikh and N. Bhat, "Housing Society Management Web Application with recommendation system," Imperial Journal of Interdisciplinary Research , vol. 3, no. 4, pp. 2184-2187, 2017.

[5] R. Bhagwat, A. Bharadwaj, V. Harsode and A. Chawake, "Society Management Application on Android," International Research Journal of Engineering and Technology, vol. 5, no. 5, pp. 2509-2511, 2018.

[6] S. Kudale, C. Amarnan, H. Sawakare and S. Kokate, "Housing Society Management," INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH VOLUME, vol. 7, no. 5, pp. 85-89, 2018.

[7] H. Solanki, D. Yadav and A. Yadav, "Society Management System," IOSR Journal of Computer Engineering, vol. 21, no. 2, pp. 1-4, march-april 2019.

[8] A. Shekhar, "blog.mindorks.com," MindOrks, 4 march 2020. [Online]. Available: <https://blog.mindorks.com/mvvm-architecture-android-tutorial-for-beginners-step-by-step-guide>.

[9] T. Lou, "A Comparison of Android Native App Architecture - MVC, MVP and MVVM," Eindhoven university of technology, 2016.