

## AMG : College Management App

**MR.ANURAG SHRIRANG DEWARDE**

*Ashokrao Mane Group Of institution,vathar  
Department of Computer Science and Engineering*

[anuragdewarde@gmail.com](mailto:anuragdewarde@gmail.com)

**MR.PRASAD DILIP PATIL**

*Ashokrao Mane Group Of institution,vathar  
Department of Computer Science and Engineering*

[prasadpatil8085@gmail.com](mailto:prasadpatil8085@gmail.com)

**MR.OMKAR ANANDRAO GAIKWAD**

*Ashokrao Mane Group Of institution,vathar  
Department of Computer Science and Engineering*

[omigaikwad2104@gmail.com](mailto:omigaikwad2104@gmail.com)

**MR.SARTHAK ARUN MAGDUM**

*Ashokrao Mane Group Of institution,vathar  
Department of Computer Science and Engineering*

[sarthakmagdum3@gmail.com](mailto:sarthakmagdum3@gmail.com)

**MR. DHAIRYASHEEL VIJAY PATIL**

*Ashokrao Mane Group Of institution,vathar  
Department of Computer Science and Engineering  
[dhairyasheelpatil564@gmail.com](mailto:dhairyasheelpatil564@gmail.com)*

**PROF. PRACHI G. CHAVAN**

*Assitant Professor at Ashokrao Mane Group Of  
institution,vathar  
Department of Computer Science and Engineering  
[pgc@amgoi.edu.in](mailto:pgc@amgoi.edu.in)*

\*\*\*

**Abstract** - The traditional college management process often relies on manual paperwork, fragmented communication systems, and offline attendance tracking methods, which lead to inefficiency, data redundancy, delayed information sharing, and increased administrative workload. To address these challenges, this paper presents **AMG: College Management App**, a mobile-based academic management system developed for educational institutions to streamline communication, attendance management, academic updates, and administrative operations through a centralized digital platform.

The proposed system is developed using **Jetpack Compose** for the Android frontend, **Flask RESTful API** for backend services, and **MySQL** as the centralized database management system. The application follows a role-based architecture consisting of **student**, **teacher**, and **administrator** modules, enabling secure and organized access to academic functionalities. Students can access attendance records, announcements, achievements, and events, while teachers can manage attendance, publish updates, and maintain academic records through dedicated interfaces.

The system integrates REST API communication and cloud-hosted database services to ensure efficient real-time data synchronization and scalable performance. Additionally, the application adopts the **MVVM architectural pattern** with repository and state management mechanisms to improve maintainability, responsiveness, and modular development. Experimental usage of the system demonstrates reduced paperwork, faster academic communication, centralized record

management, and improved accessibility for both students and faculty members.

The proposed solution provides an efficient, scalable, and user-friendly digital ecosystem for modern educational institutions and establishes a strong foundation for future enhancements such as AI-based analytics, smart attendance systems, push notifications, and online learning integration.

### 1.INTRODUCTION

This The rapid growth of digital technology has transformed many sectors, but several educational institutions still rely on traditional methods for managing academic and administrative activities. Manual handling of student records, announcements, institutional updates, and communication often leads to increased paperwork, delayed information sharing, and inefficient data management.

To overcome these challenges, educational institutions require a centralized and user-friendly digital platform capable of managing multiple services efficiently. Mobile applications provide an effective solution by offering real-time access to information and improved communication among students, teachers, parents, and administrators.

This paper presents **AMG: College Management App**, a mobile-based institutional management system developed to simplify academic and administrative operations through a unified Android application. The system is developed using **Jetpack Compose** for the frontend, **Flask RESTful API** for backend services, and **MySQL** as the centralized database system. The application includes multiple modules such as **student**, **teacher**, **parent**, **guest**, and **data entry** sections to provide role-based access and secure management of institutional data.

The proposed system improves communication efficiency, centralizes institutional information, reduces manual work, and provides a scalable digital ecosystem suitable for modern educational institutions.

## 2. LITERATURE REVIEW:

In recent years, the education sector has experienced significant digital transformation through the adoption of mobile applications, cloud computing technologies, and centralized management systems. Educational institutions are increasingly shifting from traditional paper-based processes to digital platforms in order to improve academic administration, communication efficiency, and institutional data management. Several researchers and developers have proposed various college and school management systems to simplify operations such as student record handling, notice distribution, timetable management, fee processing, and communication between faculty members and students.

Traditional college management methods mainly depend on manual paperwork and disconnected systems for handling academic and administrative tasks. These methods often lead to data redundancy, delayed communication, record maintenance difficulties, and increased workload for institutional staff. In many cases, important information such as announcements, event details, academic updates, and departmental activities are not efficiently delivered to all users due to the lack of a centralized communication platform.

Existing educational management systems mainly focus on specific functionalities such as attendance tracking, examination management, or online notice boards. Although these systems help reduce manual work, many of them lack integration between different institutional modules. Several applications are web-based only and do not provide proper mobile accessibility, which limits real-time access to institutional information for students, teachers, and parents. Some systems also face challenges related to scalability, user experience, data synchronization, and secure role-based access management.

With the advancement of Android application development and REST API technologies, modern educational platforms are becoming more flexible, scalable, and user-friendly. Mobile-based applications allow users to access academic information anytime and anywhere using smartphones. Technologies such as Jetpack Compose, cloud-hosted backend servers, and

centralized relational databases provide efficient solutions for managing institutional operations digitally.

Researchers have also emphasized the importance of centralized systems that can support multiple user roles within a single application. Parent interaction systems, guest information portals, and administrative data handling modules are becoming increasingly important in modern educational environments. However, many currently available systems either provide limited functionality or require separate platforms for different operations, resulting in fragmented user experiences.

The proposed **AMG: College Management App** aims to overcome these limitations by integrating multiple institutional services into a single Android-based platform. The system includes dedicated modules for students, teachers, parents, guests, and data entry operators, enabling organized and secure access to institutional information. The application is developed using **Jetpack Compose** for frontend design, **Flask RESTful API** for backend communication, and **MySQL** as the centralized database management system.

Unlike many existing systems, the proposed application focuses on centralized communication, role-based functionality, real-time data access, and scalable architecture. The integration of modern Android development practices and cloud-based backend services helps improve maintainability, responsiveness, and overall system performance. Therefore, the AMG College Management App provides an efficient and user-friendly digital ecosystem suitable for modern educational institutions.



Fig -1: App Illustration

### 3. PROPOSED SYSTEM

#### 3.1 System Description :

The proposed system, **AMG: College Management App**, is a mobile-based institutional management platform developed to simplify and digitize various academic and administrative operations within educational institutions. The system provides a centralized environment where students, teachers, parents, guests, and administrative staff can access institutional services through role-based interfaces.

The application is developed using **Jetpack Compose** for Android frontend development, **Flask RESTful API** for backend communication, and **MySQL** as the centralized database management system. The system follows a client-server architecture that enables secure communication between the mobile application and backend server using REST APIs.

Unlike traditional systems that rely on paperwork and separate platforms for different operations, the proposed system integrates multiple institutional services into a single mobile application. This improves communication efficiency, reduces manual work, centralizes academic information, and enhances accessibility for all users.

The system is designed to provide a scalable, secure, and user-friendly digital ecosystem suitable for modern educational institutions.

#### 3.2 Functional Modules

##### 1) Student Module

The Student Module is designed to provide students with centralized access to academic and institutional information through a mobile-friendly interface. Students can view announcements, achievements, departmental updates, institutional events, and other important notifications directly from the application. The module improves communication between students and the institution by reducing dependency on physical notice boards and manual information sharing methods.

The student interface is developed using Jetpack Compose to provide a modern, responsive, and user-friendly experience. The module also helps students stay updated with college activities in real time through API-based data synchronization with the backend server.

##### 2) Teacher Module

The Teacher Module allows faculty members to manage and publish academic and institutional information efficiently. Teachers can upload announcements, update departmental activities, share event details, and manage information visible to students through the centralized system.

The module acts as an important communication bridge between faculty members and students. By digitizing information management processes, the system reduces

paperwork and improves the speed of academic communication within the institution.

The teacher module also provides secure role-based access so that only authorized faculty members can perform content management operations.

##### 3) Parent Module

The Parent Module is developed to strengthen communication between parents and the educational institution. Through this module, parents can stay informed about institutional activities, announcements, student-related updates, achievements, and important college events.

The module helps improve transparency and parental involvement in academic activities by providing centralized access to information through a simple mobile interface. This reduces communication gaps between institutions and guardians while ensuring easy accessibility of important updates.

##### 4) Guest Module

The Guest Module allows external users and visitors to explore general institutional information without requiring authentication. This section provides details about departments, campus facilities, achievements, events, infrastructure, and other public information related to the institution.

The guest section acts as a digital information portal for new students, visitors, and external users interested in learning about the college. By providing organized and easily accessible information, the module improves institutional visibility and user engagement.

##### 5) Data Entry Module

The Data Entry Module is developed for administrative staff responsible for maintaining institutional records and managing centralized information. This module allows authorized users to add, update, and manage important data stored within the system database.

The module simplifies administrative operations by digitizing record management processes and reducing manual paperwork. It also improves data consistency and accuracy by ensuring that information is maintained through centralized database operations.

The data entry section plays an important role in keeping the institutional data updated and synchronized across all application modules.

##### 6) Authentication and User Management Module

The Authentication and User Management Module handles secure login functionality and role-based access control within the application. Different users such as students, teachers, parents, guests, and data entry operators are provided access according to their assigned permissions and responsibilities.

The module ensures secure communication between the frontend application and backend server using API-based authentication mechanisms. It also helps protect institutional data from unauthorized access and maintains user privacy within the system.

## 7) Backend Management Module

The Backend Management Module acts as the central processing unit of the system. The backend is developed using Flask RESTful API and is responsible for handling client requests, processing data, managing API communication, and interacting with the MySQL database.

Whenever a user performs an operation within the application, the request is processed by the backend server, which retrieves or updates the required data from the database and sends the response back to the frontend application.

The backend system ensures efficient data synchronization, centralized processing, scalability, and smooth communication between all modules of the application.

## 8) Database Management Module

The Database Management Module uses MySQL as the centralized database system for storing institutional records and application data. The database contains information related to announcements, achievements, user profiles, departmental details, events, and other institutional content.

The module ensures proper data organization, fast retrieval, secure storage, and reliable management of institutional information. Centralized database management also improves scalability and simplifies future system enhancements.

## 3.3 System Workflow:

The system operates using a client-server architecture where users interact with the Android application through role-based interfaces. Users first access the application and navigate to their respective modules such as student, teacher, parent, guest, or data entry sections.

When a user performs an operation, the request is sent from the Android application to the Flask backend server through REST APIs. The backend processes the request, communicates with the MySQL database, and retrieves or updates the required information.

After processing, the server sends the response back to the application, where the data is displayed to the user in a structured and user-friendly format. This workflow ensures centralized data management, secure communication, and efficient synchronization between different modules of the system.

## 4. SYSTEM ARCHITECTURE

The AMG College Management App follows a multi-layered client-server architecture designed to provide secure, scalable,

and efficient communication between different system components. The architecture separates the application into frontend, backend, and database layers, enabling modular development, centralized data management, and simplified maintenance.

The system architecture is mainly divided into the following layers:

### 4.1 Frontend Layer

The frontend layer represents the Android application developed using **Jetpack Compose**. This layer provides interactive user interfaces for different modules such as student, teacher, parent, guest, and data entry sections.

The frontend is responsible for:

- Collecting user input
- Displaying institutional information
- Managing navigation between screens
- Sending API requests to the backend server
- Presenting data in a user-friendly format

The use of Jetpack Compose improves UI responsiveness, maintainability, and modern mobile application design.

### 4.2 Backend Layer

The backend layer is developed using **Flask RESTful API** and acts as the central processing unit of the system. It handles communication between the Android application and the database.

The backend performs the following operations:

- API request handling
- User authentication and authorization
- Data processing and validation
- Database communication
- Data synchronization
- Role-based access management

The backend ensures secure and efficient transfer of data between different modules of the application.

### 4.3 Database Layer

The database layer uses **MySQL** as the centralized database management system. It stores all institutional data including:

- User information
- Announcements
- Achievements
- Department details
- Events
- Parent data

- Guest information
- Administrative records

The database layer ensures secure storage, efficient retrieval, and proper management of institutional records.

## 4.4 API Communication Layer

The API Communication Layer enables interaction between the frontend application and backend server through REST APIs. Whenever a user performs an operation in the application, API requests are sent to the backend server for processing.

This layer is responsible for:

- Sending and receiving JSON data
- Managing client-server communication
- Real-time data synchronization
- Secure data exchange

The use of REST APIs improves scalability and allows smooth integration between different components of the system.

## 4.5 Cloud Hosting Layer

The system utilizes cloud hosting services for backend deployment and database management. The Flask backend server is hosted on **Render**, while the MySQL database is hosted using **Aiven** cloud services.

Cloud integration provides:

- Remote accessibility
- Improved scalability
- Centralized data access
- Reliable server availability
- Efficient database management

This architecture enables users to access institutional services from different locations through internet connectivity.

## 4.6 System Workflow

The workflow of the AMG College Management App begins when a user accesses the Android application and logs into the system according to their role. The user interacts with different modules such as student, teacher, parent, guest, or data entry sections.

When a request is made, the frontend sends an API request to the Flask backend server. The backend processes the request, communicates with the MySQL database, and retrieves or updates the required information.

After processing, the response is returned to the Android application where the information is displayed to the user in a structured format. This workflow ensures centralized

management, secure communication, and efficient synchronization between all modules of the system.

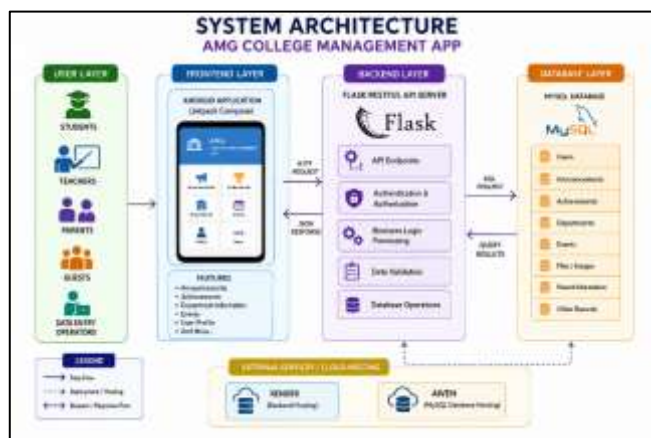


Fig 4.6 System Architecture

## 4.7 DFD Diagram:

### 4.7.1 Level 0 DFD:

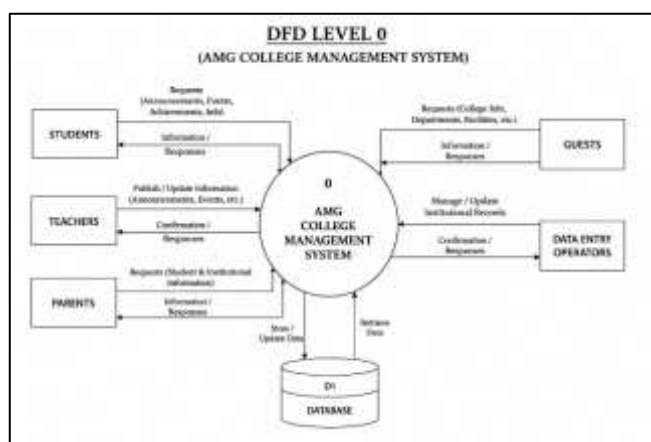


Fig 4.7.1 Level 0 Data Flow Diagram

### 4.7.1 Level 0 DFD:

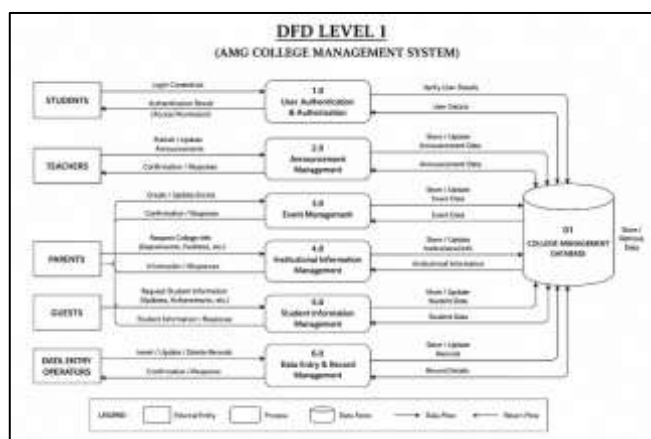


Fig 4.7.2 Level 1 Data Flow Diagram

### 4.7.1 Level 0 DFD:

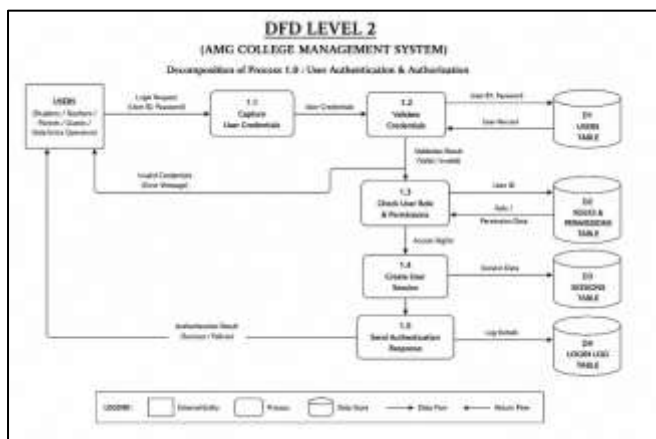


Fig 4.7.3 Level 2 Data Flow Diagram

## 5. METHODOLOGY

The AMG College Management App is developed as a centralized mobile-based platform designed to simplify academic and administrative operations within educational institutions. The system follows a client-server architecture where different modules are connected through a centralized backend server.

The Android application is developed using Jetpack Compose and provides separate role-based interfaces for students, teachers, parents, guests, and data entry operators. Users can access institutional information and services according to their assigned permissions.

When a user performs an operation, the application sends an API request to the Flask RESTful backend server. The backend processes the request, validates user access, communicates with the MySQL database, and returns the required response to the application.

The student module provides announcements, achievements, departmental information, and event updates. The teacher module allows faculty members to manage academic information and publish updates. The parent module helps guardians stay informed about institutional activities, while the guest module provides public college information. The data entry module manages centralized institutional records.

The backend server handles API communication, authentication, data processing, and synchronization between different modules. All data is stored in the MySQL database to ensure centralized management, secure storage, and efficient retrieval of information.

Cloud hosting services such as Render and Aiven are used for backend and database deployment, providing scalability, remote accessibility, and reliable system performance.

## 6. IMPLEMENTATION

The AMG College Management App is implemented as a mobile-based institutional management system using modern Android and backend technologies. The frontend of the application is developed using **Jetpack Compose**, which provides a responsive and user-friendly interface for different user modules such as student, teacher, parent, guest, and data entry sections.

The backend system is developed using **Flask RESTful API**, which handles API communication, request processing, user authentication, and interaction with the database. REST APIs are used to establish secure communication between the Android application and backend server.

The system uses **MySQL** as the centralized database management system for storing institutional information such as announcements, achievements, departmental details, events, user profiles, and administrative records. The database ensures secure storage and efficient retrieval of information.

The application follows the **MVVM (Model-View-ViewModel)** architecture along with repository-based data handling to improve maintainability, scalability, and structured data management. Retrofit is used for API communication between the frontend and backend.

Cloud hosting services are also integrated into the system. The Flask backend server is hosted on **Render**, while the MySQL database is hosted using **Aiven** cloud services. This allows remote accessibility, centralized data management, and scalable system performance.

The complete system is tested under different user scenarios to ensure smooth functionality, secure communication, efficient data synchronization, and reliable performance across all modules of the application.

## 7. RESULTS AND DISCUSSION

The AMG College Management App was evaluated to improve institutional communication, centralized information management, and accessibility of academic services through a mobile platform. The system successfully integrates multiple modules such as student, teacher, parent, guest, and data entry sections within a single application.

The application demonstrated efficient performance in handling announcements, departmental updates, achievements, events, and institutional information through centralized database management. Role-based access control ensured secure access to different modules according to user permissions and responsibilities.

The frontend developed using Jetpack Compose provided a responsive and user-friendly interface, while the Flask RESTful backend ensured smooth API communication and reliable data synchronization with the MySQL database. The

integration of cloud hosting services such as Render and Aiven improved remote accessibility and system scalability.

The proposed system reduced manual paperwork, simplified institutional data management, and improved communication efficiency between students, teachers, parents, and administrative staff. The overall implementation demonstrated reliable performance, secure communication, and centralized management suitable for modern educational institutions.

## 8. FUTURE WORK

The AMG College Management App can be further enhanced by integrating additional advanced features and modern technologies to improve institutional management and user experience.

Future improvements may include:

- Online lecture and meeting integration for digital learning support.
- Advanced analytics dashboards for institutional reporting and monitoring.
- Multi-language support for improved accessibility.
- Biometric or face recognition authentication for enhanced security.
- Cloud file storage for sharing study materials and documents.
- Integration of placement and internship management modules.

These enhancements can improve scalability, automation, security, and overall efficiency of the system while providing a more advanced digital ecosystem for educational institutions.

## 9. CONCLUSION

The AMG College Management App is a centralized mobile-based institutional management system developed to simplify academic and administrative operations within educational institutions. The system successfully integrates multiple modules such as student, teacher, parent, guest, and data entry sections into a single Android application.

The application is developed using Jetpack Compose, Flask RESTful API, and MySQL database technologies to provide secure communication, centralized data management, and efficient system performance. The implementation of role-based access control improves security and ensures organized management of institutional information.

The proposed system reduces manual paperwork, improves communication efficiency, enhances accessibility of institutional services, and provides a scalable digital platform suitable for modern educational environments. Cloud integration and API-based communication further improve

reliability, maintainability, and remote accessibility of the system.

Overall, the AMG College Management App provides an efficient, user-friendly, and scalable solution for digitizing institutional operations and establishing a strong foundation for future technological enhancements in educational management systems.

## ACKNOWLEDGEMENT

We express our sincere gratitude to our project guide, **Prof. P. G. Chavan**, for his valuable guidance, encouragement, and continuous support throughout the development of the **AMG College Management App** and the preparation of this research paper.

We are also thankful to the Department of Computer Science and Engineering, Ashokrao Mane Group of Institutions, Vathar, for providing us with the opportunity, resources, and technical environment required for successfully completing this project.

We would like to extend our appreciation to all faculty members, friends, and classmates who directly or indirectly contributed to the successful implementation of this work through their suggestions and support.

Finally, we express our gratitude to our parents and family members for their constant motivation and encouragement throughout the project development and research work.

## 10. REFERENCES

1. [\*Ashokrao Mane Group of Institutions Official Website – Institutional information and college-related data reference used for the AMG College Management App.\*](#)
2. *Android Developers*, “Jetpack Compose Documentation,” Google Developers.
3. *Flask Documentation*, “Flask Web Development Framework,” Pallets Projects.
4. *MySQL Documentation*, “MySQL Database Management System,” Oracle Corporation.
5. *Retrofit Documentation*, “Type-Safe HTTP Client for Android,” Square Inc.
6. *Render Cloud Platform Documentation*, “Cloud Application Hosting Services.”
7. *Aiven Documentation*, “Managed MySQL Cloud Database Services.”
8. <https://github.com/AnuragDewarde/college-management-server.git> - Server Url

9. *Ian Sommerville, Software Engineering, 10th Edition, Pearson Education, 2015.*
10. *Roger S. Pressman, Software Engineering: A Practitioner's Approach, McGraw-Hill Education, 2014.*
11. *Android Developers, "MVVM Architecture Guidelines for Android Applications."*
12. *Flask-RESTful Documentation, "REST API Development using Flask."*
13. *Google Developers, "REST API Design and Client-Server Communication."*

## 11. Author's :



**Mr. Anurag Shrirang Dewarde** is currently pursuing a Bachelor's Degree in Computer Science & Engineering at Ashokrao Mane Group Of Institution, vathar. He is publishing this paper under the esteemed guidance of Prof. Prachi G. Chavan



**Mr. Prasad Dilip Patil** is currently pursuing a Bachelor's Degree in Computer Science & Engineering at Ashokrao Mane Group Of Institution, vathar. He is publishing this paper under the esteemed guidance of Prof. Prachi G. Chavan



**Mr. Omkar Anandrao Gaikwad** is currently pursuing a Bachelor's Degree in Computer Science & Engineering at Ashokrao Mane Group Of Institution, vathar. He is publishing this paper under the esteemed guidance of Prof. Prachi G. Chavan



**Mr. Sarthak Arun Magdum** is currently pursuing a Bachelor's Degree in Computer Science & Engineering at Ashokrao Mane Group Of Institution, vathar. He is publishing this paper under the esteemed guidance of Prof. Prachi G. Chavan



**Mr. Dhairyasheel Vijay Patil** is currently pursuing a Bachelor's Degree in Computer Science & Engineering at Ashokrao Mane Group Of Institution, vathar. He is publishing this paper under the

esteemed guidance of Prof. Prachi  
G. Chavan



**Prof. Prachi G. Chavan** is currently working as an Assistant Professor in Computer Science & Engineering at Ashokrao Mane Group Of Institution, vathar. she has been guiding and mentoring students in various academic and research activities. she has provided her valuable guidance for the successful completion and publication of this paper.