



Faculty and Parent Communication App for Improved Engagement in Educational Institutions

Atharva Mane¹, Niraj Patil², Ajay Khot³, Ganesh Naik⁴, Prof. Rahul Kapse⁵

¹²³⁴⁵Department of Computer Engineering, Pillai HOC College of Engineering and Technology, Khalapur, Maharashtra, India

ABSTRACT:

Effective communication between faculty and parents is essential for student growth, especially in semi-urban and rural areas where digital penetration is limited. This paper presents a mobile-based solution, the *Faculty and Parent Communication App (FPCA)*, designed using React Native and Firebase. The app ensures real-time announcements, private chats, group communication, role-based access, and privacy protection. Unlike traditional systems that depend on manual notices or paper records, FPCA leverages OTP verification, cloud-based storage, and multilingual support to enable a secure, fast, and scalable platform. This study compares our system with existing apps like Collat and School Notification App, highlighting FPCA's unique features, especially focused role controls, verified faculty onboarding, and private one-on-one interactions.

Keywords: Parent Communication, Firebase, React Native, Education App, OTP Verification, Group Messaging, School Management.

Introduction:

Parent-teacher communication has traditionally relied on physical meetings, manual notices, or inconsistent third-party apps. In rural and semi-urban India, such models suffer from poor accessibility and delayed updates. To address this, our team developed the *Faculty and Parent Communication App (FPCA)* — a dedicated mobile platform that ensures structured, role-based, and secure interaction among faculty, parents, and administrators. FPCA is tailored to empower educational institutions by providing a modern digital interface where announcements, group chats, and one-on-one interactions happen seamlessly. Built with widely-used tools like React Native and Firebase, it ensures compatibility across Android and iOS devices. The app has been designed keeping in mind the unique challenges of Indian schools, such as language diversity, internet limitations, and data privacy concerns.

Why FPCA is Needed?

- Lack of timely updates: Traditional notice boards and meetings delay important communication.
- Limited accessibility: Parents in rural areas often can't visit schools frequently due to work or distance.
- Privacy risks: Existing solutions expose personal information like faculty phone numbers.
- Unstructured messaging: Many platforms don't differentiate roles, leading to chaotic or unauthorized communication.
- Approval and verification missing: Teachers are added without admin approval in most apps, leading to security risks.

FPCA resolves these issues by offering a clean, role-based communication channel that maintains privacy, improves security, and promotes responsible academic engagement between faculty and parents.

Methodology:

The development of the FPCA app followed a systematic and iterative approach.

1. Initial Research and Review

We began by reviewing several existing projects such as Collat, School Notification App, and common Learning Management Systems (LMS). Our goal was to identify gaps in privacy, user role control, and parent interactivity. Through research papers and app store analysis, we found that existing solutions lacked secure one-on-one communication, structured faculty onboarding, and localized accessibility.

2. Planning and Design Phase

Using Figma, we designed the user interface for various roles: Parent, Faculty, and Admin. Special attention was given to dark mode compatibility and UI simplicity for rural users. The flow diagrams, registration flow, and component hierarchy were also finalized in this phase.

3. Development Process

We used React Native CLI with TypeScript for frontend development, and Firebase Modular SDKs for backend integration. The coding process followed modular structure:

- Authentication: Integrated Firebase Phone Auth with OTP handling screens.
- Firestore Structure: Created collections for users, groups, chats, announcements.
- Chat Features: Developed real-time group and private chat using listeners and queries.
- Faculty Approval Logic: Added Cloud Functions to handle approval requests, trigger notifications, and update Firestore roles.
- Notifications: Configured Firebase Cloud Messaging to send alerts on message updates and announcements.
- Media and Files: Integrated Firebase Storage to allow image and PDF sharing.

4. Testing and Optimization

We performed device testing across Android phones with limited internet to ensure performance and responsiveness. Edge cases were handled for offline states, network delays, and invalid OTP inputs. Manual testing and functional validation were done for each role.

Comparative Advantages of Existing System:

Many educational communication apps exist, such as Collat, School Notification App, ClassDojo, and Remind. However, most of these apps either cater to general educational management or are focused on western school systems. ClassDojo emphasizes classroom behavior tracking and visual engagement for students, but lacks regional customization, offline support, or OTP-based verification. Remind is widely used for school announcements and messaging, but it lacks group chat features and role-specific restrictions required in Indian school systems.

FPCA bridges these gaps with features tailored for Indian institutions, rural connectivity needs, OTP verification, and strict role-based access. Unlike Collat or School Notification App, FPCA provides structured communication, admin-verified onboarding for faculty, private one-on-one replies from group messages, and complete privacy for faculty profiles.

Feature	Collat	School Notification App	ClassDojo	Remind	FPCA (Proposed)
Role-based chat	NO	NO	NO	NO	YES
OTP Verified Faculty	NO	NO	NO	NO	YES
Group Privacy Controls	NO	NO	NO	NO	YES
Private Replies from Group	NO	NO	NO	NO	YES
No Faculty Contact Sharing	NO	NO	NO	YES	YES
Profile Management	YES	YES	YES	YES	YES
One-on-One Messaging	YES	NO	YES	YES	YES

The FPCA offers a complete, secure, and structured communication system that aligns with the specific needs of Indian schools, ensuring responsible communication and reduced information overload for all users.

FPCA improves upon existing systems by giving structured communication controls and faculty privacy while enabling meaningful interaction.

Proposed System FPCA:

- 1) User Roles and Registration
 - Startup Options: Login or Register
 - Registration with Role Selection: Faculty or Parent
 - Faculty: Requires admin (faculty) approval after registration + OTP Verification.
 - Parent: Registers with student's name and phone number.
- 2) Core Features
 - Announcements Screen: Faculty can send important updates. Parents can only view them. Faculty phone numbers remain hidden.
 - Group Chat: Faculty can form groups and post messages. Parents can only view messages.
 - Private Reply Button: Parents can respond privately to faculty from group messages.
 - One-on-One Chat: Faculty and Parent can chat directly. Parents cannot message other parents.
 - Profile Management: Users can upload profile photos; Parents see limited faculty info (name, email only).

3) Technologies Used

- React Native (CLD): Frontend with TypeScript
- Firebase (Modular SDK): Authentication (Phone OTP), Firestore, Cloud Messaging, Storage, Functions
- Google Cloud Platform
- Figma: UI/UX Design

4) Security and Privacy

- OTP verification for Faculty authentication.
- Role-based access and chat visibility.
- No faculty phone numbers visible to parents.
- Firestore rules for secure data access.

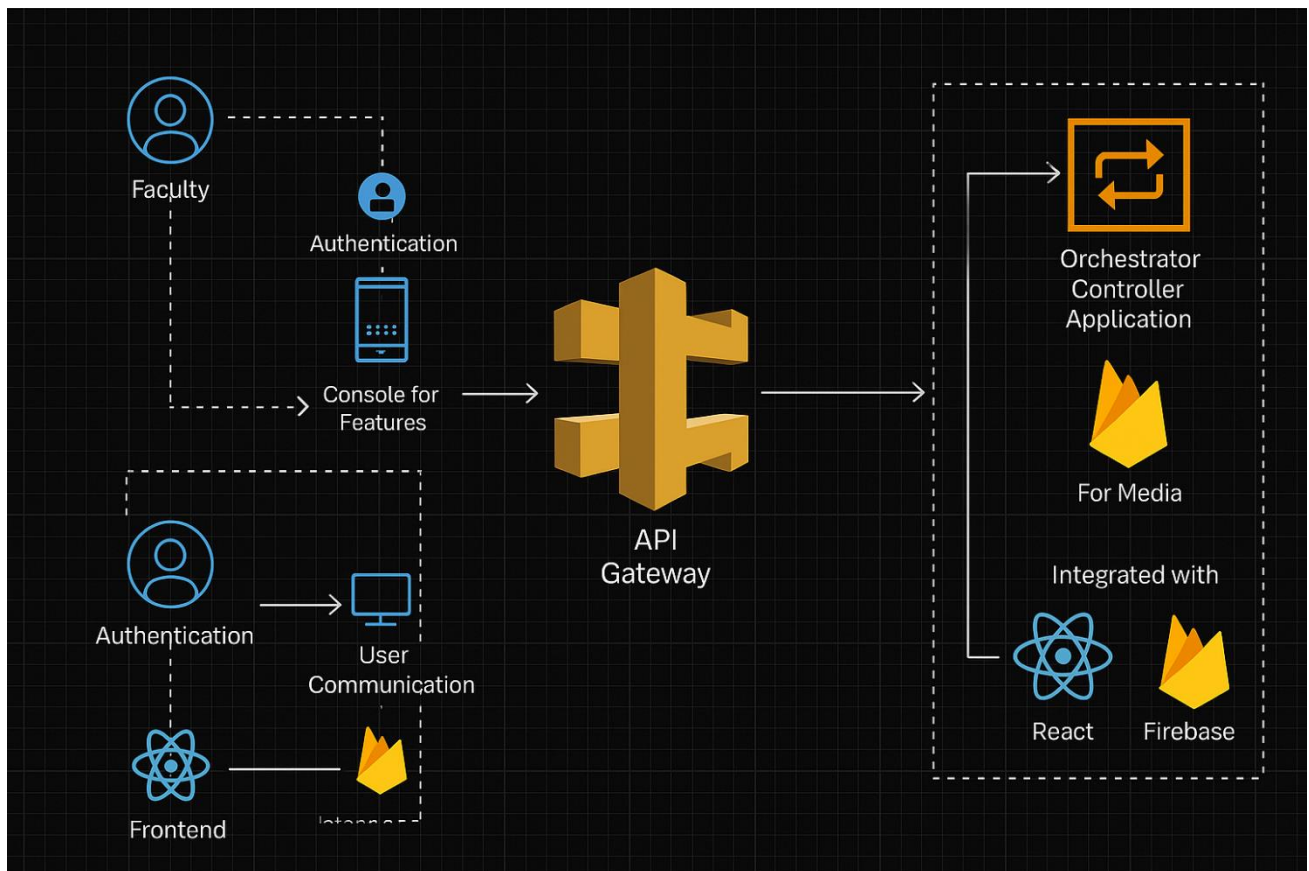
System Architecture:

Fig: System Architecture

Step-by-Step Workflow of FPCA Architecture

Step 1: User Opens the App

- Faculty or Parent launches the app (built using React Native).
- The app shows login or registration screens.

Step 2: Authentication via Firebase

- Users enter their phone number.
- OTP is sent and verified using Firebase Authentication.
- On success:
 - Faculty are checked for admin approval.
 - Parents are directly logged in.

Step 3: Role-Based Frontend Access

- Based on the user's role:
 - Faculty sees features like announcements, groups, profile, and chat.
 - Parent sees announcements and has access to private chat only.
- The app UI adapts based on role (handled using conditional logic in frontend).

Step 4: API Gateway Processing

- All requests from the app (e.g., send message, view chat, upload media) go through the API Gateway.
- The gateway:
 - Validates user session
 - Routes the request to the appropriate backend module

Step 5: Orchestrator Controller (Cloud Functions)

- Complex logic like:
 - Faculty approval
 - Notification triggering
 - Data validation and access checks
- These are handled by Firebase Cloud Functions, shown as Orchestrator Controller Application in the diagram.

Step 6: Database Interactions (Firestore)

- Based on the request, Firestore is used to:
 - Read or write chat messages
 - Save or fetch announcements
 - Store user roles and profiles

Step 7: Media Uploads (Firebase Storage)

- When faculty or parents upload images or PDFs:
 - Files are stored in Firebase Storage
 - The file URLs are saved in Firestore and shared in chats or announcements

Step 8: Push Notifications (FCM)

- When:
 - A new message is sent
 - A new announcement is posted
 - A faculty is approved
- Then Firebase Cloud Messaging (FCM) sends a push notification to the user's device

Step 9: Real-time Updates

- Firestore listeners keep the app synced:
 - New messages show instantly
 - Announcements appear live without refreshing

Step 10: Secure and Scalable Operations

- Firebase ensures:
 - End-to-end encryption
 - Secure data rules
 - Real-time performance
 - Auto-scaling for user traffic

Results

So this is the result of our hard work and detailed development process. The Faculty and Parent Communication App (FPCA) was successfully implemented with all major planned features. The app has been fully tested on Android devices and works smoothly even on low internet connections.

This section shows the actual working screens of our application, including login, OTP verification, announcements, group chats, and private messaging.

We used real Firebase services for data storage, authentication, and notifications. Our UI was created using Figma and developed using React Native CLI. All features like media sharing, dark mode, role-based access, and approval workflow are functioning as expected.

Below, we have included screenshots from the app to demonstrate:

- The clean and simple design of the login and dashboard
- Real-time chat interface for faculty and parents
- Announcements display with privacy for faculty
- Approval system for new faculty

These visuals prove that the app is working as intended and is ready for real-world use in schools and colleges.

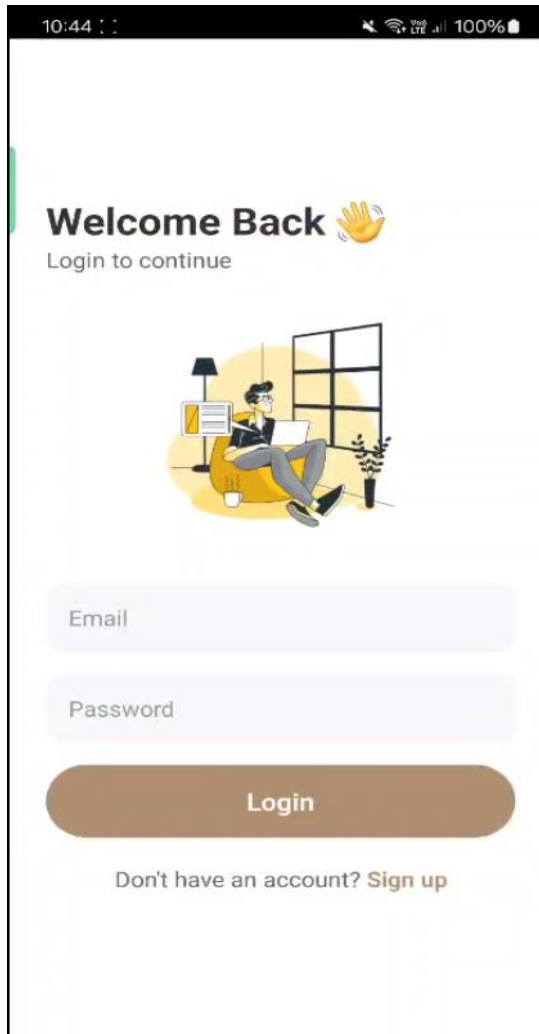


Fig: Login Page

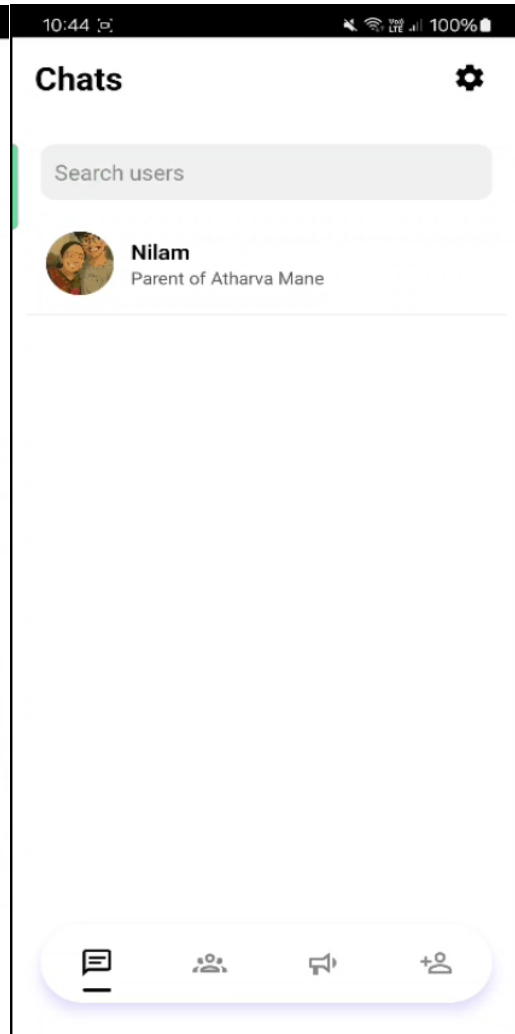


Fig: Faculty Landing page

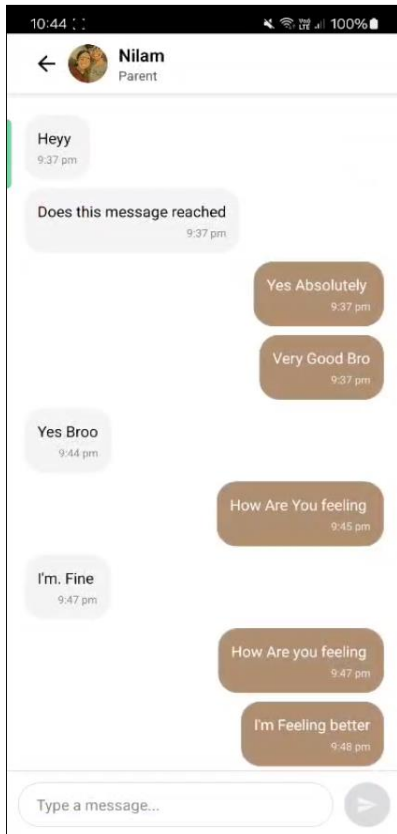


Fig: Faculty chat Screen

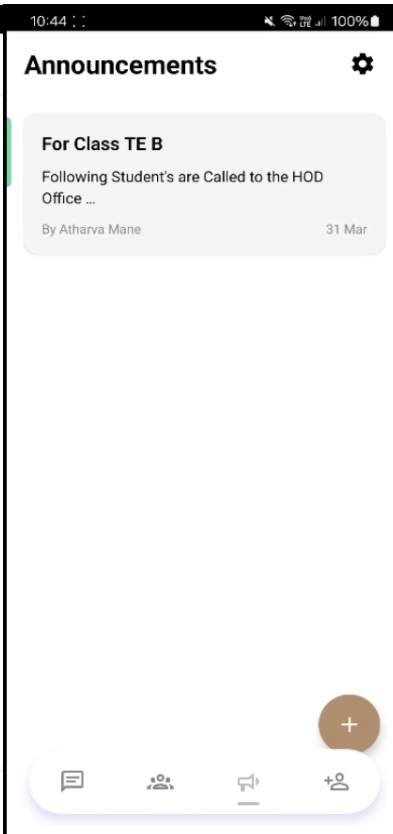


Fig: Announcement Screen

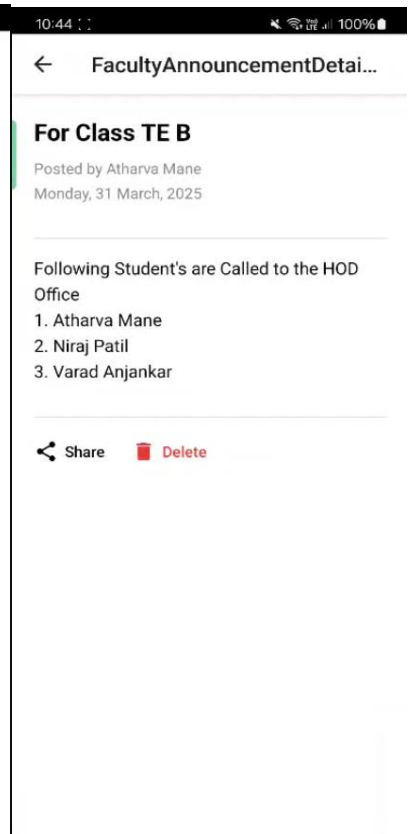


Fig: Announcement Screen

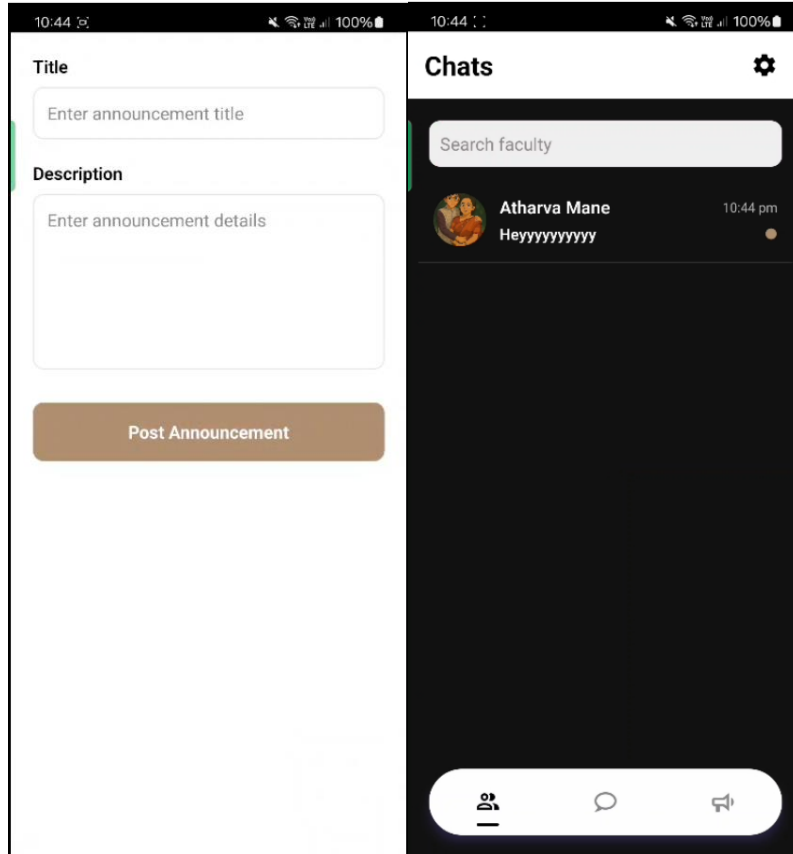


Fig: Make Announcement

Fig: Parent landing Page

Conclusion

The Faculty and Parent Communication App (FPCA) is a useful and smart solution for improving the way schools and families communicate. In many schools, especially in rural or semi-urban areas, sending notices or arranging meetings is slow and inconvenient. With FPCA, this problem is solved by allowing instant communication between faculty and parents through a mobile app.

The app ensures that only verified faculty can send messages or make announcements. Parents can reply privately, view updates, and stay informed without needing to visit the school physically. All personal data is kept private, especially the phone numbers of faculty members.

We used trusted tools like React Native for app development and Firebase for handling data, login, and notifications. This makes the app fast, secure, and easy to scale. It works well even on low internet connections, which is perfect for rural areas.

Compared to other apps like ClassDojo, Remind, or School Notification App, FPCA provides better control, privacy, and local-friendly design. It also includes special features like OTP login, admin approval, and media sharing in a way that fits Indian education needs.

In the future, we plan to add more features like offline access, AI-based reports, automatic attendance tracking, and better dashboards for admin users. Our goal is to make school communication smooth, secure, and accessible for everyone involved.

Overall, FPCA is not just an app, but a step forward in connecting homes and classrooms in a better, smarter, and safer way.

REFERENCES:

List all the material used from various sources for making this project proposal

Research Papers:

1. Dhomse V., Bhoite D., Mishra S., Singh P., Sharif I. (2023). *School Notification App*, IJRPR Vol 4, No 3, pp. 799–801.
2. Jaiswal A., Singh B., Jaiswal A., Srivastava A., Shahi A. (2023). *Collat – An Android Based College Application*, IJRPR Vol 4, No 5, pp. 6304–6314.
3. Sherief S.R., Sarwar M.A. (2024). *A Study of Relationship between Parent's Involvement and Student's Success*, IJRPR Vol 5, No 4, pp. 4762–4767.
4. Hashim N.M.Z., Mohamed S. (2013). *Use of Cloud Systems in Smart School Management*, International Journal of Science and Research (IJSR), ISSN 2319-7064.
5. Sharma D., Dand H. (2019). *Firebase as BaaS for College Android Applications*, International Journal of Computer Applications, Vol 178(20).
6. Joshi L.M. (2015). *A Research Paper on College Management System*, International Journal of Computer Applications, Vol 122.11, pp. 32–44.

Other:

1. Firebase Documentation - <https://firebase.google.com/docs>
2. React Native Documentation - <https://reactnative.dev>
3. Firebase Documentation - <https://firebase.google.com/docs>
4. React Native Documentation - <https://reactnative.dev>
5. IJRPR Formatting Guidelines - www.ijrpr.com (2023). *School Notification App*, IJRPR Vol 4, No 3, pp. 799–801.