VonClick

Tanvi Jaware

tanvi.jaware@vpt.edu.in,
Department of Information Technology
Vidyalankar Polytechnic
Mumbai, India

Shruthi Yadav

shruthi.yadav@vpt.edu.in,
Department of Information Technology
Vidyalankar Polytechnic
Mumbai, India

Keertika Nadar

keertika.nadar@vpt.edu.in,
Department of Information Technology
Vidyalankar Polytechnic
Mumbai, India

Shraddha Shiraskar

shraddha.shiraskar@vpt.edu.in,
Department of Information Technology
Vidyalankar Polytechnic
Mumbai, India

Kalyani Pawar

Kalyani.pawar@vpt.edu.in,
Department of Information Technology
Vidyalankar Polytechnic
Mumbai, India

Ypgita Khandagale,

yogita.khandagale@vpt.edu.in,
Department of Information Technology
Vidyalankar Polytechnic
Mumbai, India

Abstract—This is an application which helps athletes, who have represented and performed at district, state, national or international level. Central and State Governments offer Sports Quota for various government jobs like police, defence, officer, and physical training instructor, etc. A number of the reservations are available at railway and banking sectors, too. This app lists all types of government jobs available for such Indian Athletes under their quota and displays the jobs relevant to the individual user based on their profile. Thus, athletes can use this app as a one stop shop to look for a post athletic career through this app.

I. INTRODUCTION

The Facility Management App aims to optimize the management of facilities by providing an intuitive platform for tracking maintenance requests, managing assets, and monitoring inventory. The application is designed for facility managers, maintenance teams, and tenants, ensuring seamless communication and efficient operations.
The existing system for finding jobs that reserve a sports quota for Indian athletes is not as efficient and effective as our new job search app.
Firstly, the existing system requires athletes to navigate through multiple government websites and job portals to find suitable job opportunities, which can be time-consuming and inefficient. In contrast, our app provides a centralized platform that simplifies the job search process for athletes and saves them significant time and effort.

Secondly, the existing system lacks transparency and provides limited information about the job opportunities, which makes it difficult for athletes to identify suitable job opportunities that align with their skills, experience and most importantly - quota. Our app addresses this problem by providing detailed information about the job opportunities, including eligibility criteria, job descriptions, and application processes, which helps athletes make informed decisions about their career choices.

The big daddy of all the problems that we discussed in our analysis, had to be the amount of time and effort they has to spend in getting a job.
We found out that Android Studio with Java as the language would be the perfect technology for our project to be built on. Then we discussed with our mentor to confirm that we would implement database connectivity with Firebase's FireStore server.

The app proposed focuses at filtering out the jobs entered in the database by the admin and displaying it to the user that will suit his profile.

- The application provides a list of jobs (amongst other features) based on the profile that the user can choose from according to his likings.
- It satisfies the requirement of users.
- Understanding the working of system is not complicated in case of both the users and the admin.

- Have a good UI.
- Always having room for expansion and scalability

II. PROBLEM STATEMENT

Students in educational institutions face difficulties in efficiently reporting facility issues, such as broken air conditioning or maintenance needs. Current methods are often slow and lack transparency, leading to delays in problem resolution and decreased satisfaction. This project aims to develop a mobile application that allows students to quickly report issues by uploading pictures and descriptions, improving communication and efficiency in facility management.

III. PROPOSED SYSTEM

Our Facility Management App simplifies issue reporting and enhances communication between students, facility managers, and maintenance teams. Students can quickly report issues by uploading images and descriptions, while real-time tracking ensures transparency. The system categorizes reports, assigns tasks efficiently, and sends updates. With cloud-based storage and analytics, institutions can optimize maintenance and improve response times, ensuring a well-maintained environment.

IV. TECHNOLOGIES USED IN PROJECT

1. **XML in UI Design**
   Defines app layout separately from Java logic.
   Structures UI components like buttons, text fields, and images.
2. **LinearLayout Overview**
   Arranges elements in vertical or horizontal order.
   Simple, lightweight, and easy to use.
3. **Key Features**
   Orientation: Vertical (android:orientation="vertical") or Horizontal (android:orientation="horizontal").
   Weight: Distributes space among elements (android:layout_weight).
   Gravity: Aligns elements (android:gravity, android:layout_gravity).
4. **Uses in Facility Management App**
   Login Screen: Aligns text fields and buttons.
   Maintenance Form: Structures input fields and submit button.
   Dashboard & Profile Screens: Organizes user details and actions.
5. **Advantages**
   Simple and efficient.
   Ensures consistent UI alignment.
   Uses minimal system resources.

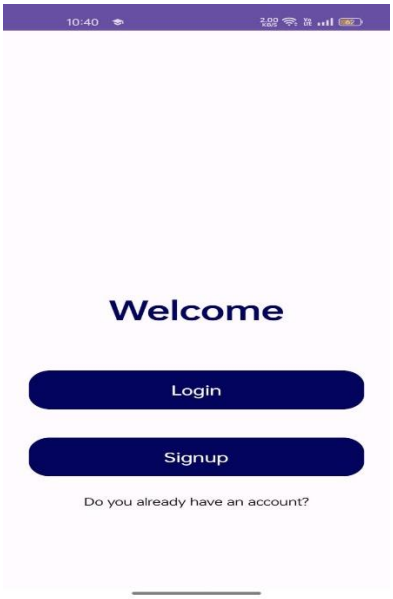
V. PLAN OF WORK

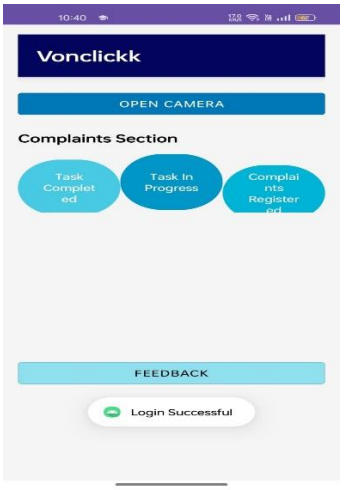
Software development life cycle:

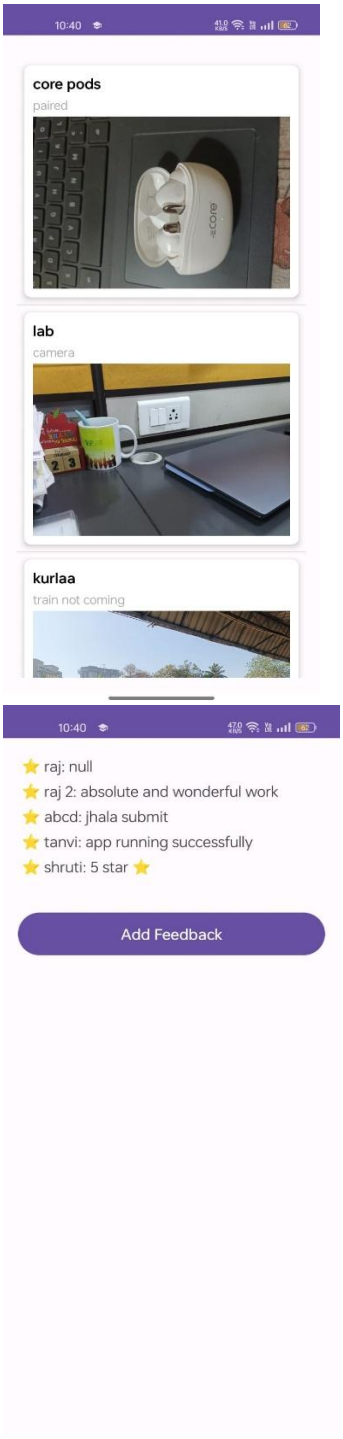
The image is a circular Software Development Life Cycle (SDLC) diagram representing the different phases of software development. It has a black-and-white, flowchart-style design with icons and text describing each phase. The central element of the diagram is a smartphone, symbolizing the mobile application being developed. Arrows connect the different phases, indicating the flow of the development process.

Planning – Illustrated with a document labeled "SDD" (Software Design Document) and a checklist icon.
Requirement Analysis / Coding – Shown with a tablet displaying a form, representing requirement gathering.
Design – Depicted with a notepad, gears, and a checklist, signifying UI/UX and system design.
Implementation / Coding – Represented by a laptop and gears, indicating the coding phase.
Testing – Illustrated with a web-based testing environment, ensuring the app functions correctly.
Deployment – Displayed with a gear icon, symbolizing the release of the application.

VI. "VONCLICK" IN YOUR PHONE







,a chat option, allowing users to prompt the AI for information through natural language interactions. Text-to-speech and vice versa functionalities enhance accessibility, providing a seamless flow of communication between the user and the AI model.

VII. SOFTWARE REQUIREMENTS

1. **Frontend (User Interface)**
   Programming Language: Java
   UI Design: XML with LinearLayout
   Development Environment: Android Studio
   Mobile OS Support: Android (API Level 21 and above)
2. **Backend (Database & Server)**
   Database: Firebase Firestore (for real-time data storage)
   Authentication: Firebase Authentication (for user login & sign-up)
   Storage: Firebase Storage (for uploading images of facility issues)
   Cloud Functions: Firebase Cloud Functions (for notifications & automation)
3. **Other Tools & APIs**
   Google Maps API: For location-based issue reporting (if needed)
   Firebase Cloud Messaging (FCM): For push notifications

Construction and Working:

VIII. MODULES

User Registration & Authentication -Users can securely Sign Up or Log In using their email, phone number, or Google authentication to ensure a safe and verified user base. Once the authentication process is completed, users are automatically redirected to the Home Page, where they can access the main features of the application.

Home Page -The Home Page serves as the central hub where users can access all key features of the application. It displays a categorized list of tasks, including Complete Tasks (resolved issues), Requested Tasks (newly reported issues awaiting action), and Progress Tasks (ongoing maintenance work by facility managers). Additionally, users can utilize the Camera Access feature to capture and upload images of issues directly. A Feedback Form is also available, allowing users to share their experiences and report any app-related problems.

Camera & Issue Reporting -Users can capture and upload pictures of maintenance issues directly through the app. Along with the images, they can provide a brief description to explain the problem in detail. Additionally, location tagging allows users to specify the exact area where the issue exists, helping facility managers respond more efficiently.

Feedback Form - The Feedback Form allows users to share their experiences, provide suggestions, and report any bugs or technical issues they encounter while using the app. This helps improve the application by gathering user input for future enhancements and troubleshooting.

In-app Notifications & Status Update - Users receive real-time notifications regarding the status of their reported issues. Task progress is updated as it moves through different stages: Requested → In Progress → Completed. Additionally, users are notified of admin updates and maintenance alerts, ensuring they stay informed about the resolution process and any important announcements.

AutoPush Notifications - The app automatically sends real-time push notifications to keep users informed about important updates. Notifications are triggered when a new issue is assigned, an issue is resolved, or when there are important app updates or announcements, ensuring users stay

updated on the maintenance process without needing to check manually.

IX. PRACTICAL APPLICATIONS

Classroom Maintenance – Students and faculty can report issues like broken projectors, damaged furniture, or faulty ACs.
Library Management – Ensures timely repairs of seating arrangements, lighting, or damaged bookshelves.
Hostel & Dormitory Maintenance – Helps students report issues related to plumbing, electricity, or room maintenance.
Campus Infrastructure – Tracks repairs for roads, pathways, elevators, and common areas.
Laboratory & Equipment Repairs – Ensures lab equipment is in working condition by enabling students to report malfunctions.
Canteen & Food Facility – Allows students to report hygiene issues, food quality concerns, or broken equipment.
Sports Complex & Gym Maintenance – Helps in tracking repairs for gym equipment, sports courts, or stadium seating.
Restroom Hygiene & Water Supply – Ensures timely maintenance of washrooms and water supply facilities.
Security & Safety Issues – Enables students to report broken CCTV cameras, non-functional emergency exits, or safety hazards.

X. CONCLUSION

The Facility Management System App is designed to make college maintenance more efficient and transparent. By allowing students and staff to report facility-related issues instantly through photos and descriptions, the app ensures quick communication with the maintenance team. This reduces delays in problem resolution and improves the overall campus experience.
With features like real-time tracking, push notifications, and an easy-to-use interface, the app streamlines facility management. In the future, integrating AI, IoT, and predictive maintenance can further enhance its effectiveness, making college infrastructure management smarter and more proactive. This app will play a crucial role in maintaining a well-functioning and hassle-free campus environment.

BIBLIOGRAPHY

[1]https://developer.android.com/studio/intro.
[2] https://docs.oracle.com/javase/8/docs/.
[3]https://docs.oracle.com/javase/tutorial/.