Parental Control Android App

Minor project Report

Submitted in Partial Fulfilment of the Requirements for the degree of

Bachelor of Technology

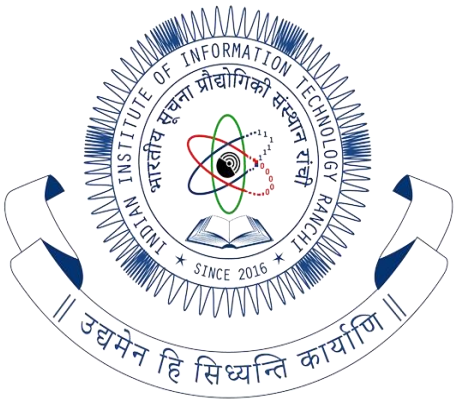
in

Department of Computer Science Engineering

Samsoth Sanjay (2021UG1090)

Under the Guidance of

Dr. Kirti Kumari



Computer Science and Engineering Department

Indian Institute of Information Technology, Ranchi

Ranchi – 835217

CANDIDATE DECLARATION

I hereby certify that the work presented in the Minor Project Report entitled Parental Control Android App, which has been submitted for the partial fulfilment of the award of degree of Bachelor of Technology in Computer Science and Engineering at Indian Institute Information Technology Ranchi is a record of my work carried out under the supervision of Dr. Kirti Kumari**.** I have cited the sources of the text(s)/figure(s)/table(s) referred for the same.

DATE: 03/12/2024 Samsoth Sanjay

PLACE: Ranchi (2021UG1090)

CERTIFICATE

This is to certify that the above statement made by the candidates is correct to the best of my knowledge and belief.

(Dr. Kirti Kumari)

Faculty

Dept. of CSE, IIIT Ranchi

Table of Contents

S.No	Contents	Page No.
1	Abstract	4
2	Introduction	5
3	Literature Survey	6
4	Features	7
5	Technology Stack	8-9
6	System Architecture	10-11
7	Implementation Details	12-13
8	Keyword and Restricted App Monitoring	14-15
9	Notifications and Alerts	16-17
10	Screenshots with Explanations	18-27
11	Use Cases	28-29
12	Conclusion	30
13	References	31

Abstract

The **Parental Control Application** is designed to assist parents in monitoring and regulating their child's mobile device usage. With a focus on simplicity and effectiveness, the application enables real-time tracking of restricted apps and keywords, ensuring that children are shielded from inappropriate content. The app leverages accessibility services to monitor suspicious activity and provides instant notifications and email alerts for any violations.

The primary objective of this project is to offer a user-friendly interface for parents to monitor and manage their child's digital activities. Through features like keyword detection, restricted app monitoring, and seamless notifications, the application promotes a safer digital environment for children.

Developed using **Android Studio** and integrated with **Spring Boot backend services**, the app demonstrates an effective combination of frontend and backend technologies to deliver robust functionality. This document outlines the app's development process, key features, implementation details, and use cases, culminating in a practical solution to address the challenges of digital parenting.

Fig 1: Parental Control Concept Visualization

Introduction

The rapid growth of technology has brought both opportunities and challenges, especially when it comes to ensuring the safety and productivity of device usage. With the increasing dependency on digital devices among children and teens, the need for a robust parental control system has become essential.

This project, titled "Parental Control Application," aims to address this need by providing parents with a comprehensive solution to monitor and regulate their children's device usage. The application integrates advanced features such as keyword monitoring, restricted app detection, and notification alerts to help parents ensure safe and appropriate usage.

The development of this application focuses on user-friendly design and efficient functionality, utilizing the latest technologies like Android SDK, Firebase Authentication, and a custom backend service built with Spring Boot. The goal is to create an intuitive tool that empowers parents while respecting the privacy and autonomy of the users.

This report outlines the objectives, design, implementation, and functionality of the Parental Control Application. Through this project, we aim to demonstrate how technology can be leveraged to create practical solutions for real-world problems.

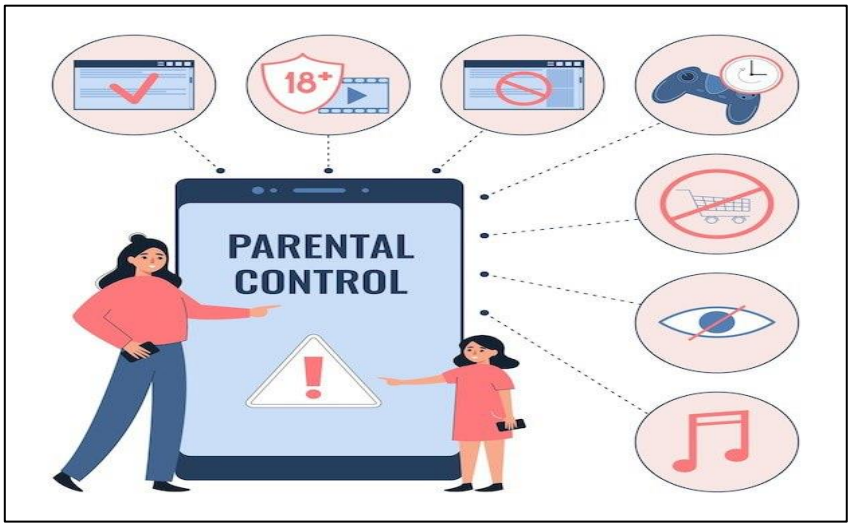


Fig 2: Parental Control Features

Literature Survey

Parental control applications have evolved over the years, driven by the growing need for child safety and digital well-being. The literature and existing solutions in this domain highlight several key areas:

1. **Existing Applications:** Popular parental control apps such as **Google Family Link**, **Qustodio**, and **Net Nanny** provide features like screen time monitoring, content filtering, and location tracking. While these applications are widely used, many lack granular keyword monitoring or real-time app usage alerts, leaving a gap for more customized control mechanisms.
2. **Keyword Monitoring:** Research studies have shown that keyword monitoring is an effective way to detect potential harmful activities, such as exposure to inappropriate content or online harassment. However, implementing such features often faces challenges in accuracy and language diversity, which we addressed by including multi-language support (e.g., Telugu and English).
3. **Technology and Privacy Concerns:** Studies emphasize the importance of balancing monitoring capabilities with user privacy. While parental control apps provide critical safety measures, they must avoid excessive intrusiveness. Our project adheres to this principle by enabling lightweight, focused monitoring while avoiding unnecessary data collection.
4. **Notifications and Alerts:** Real-time notifications are crucial for immediate action. Existing systems often delay or fail to notify parents promptly when suspicious activity is detected. Our solution addresses this limitation by integrating real-time notifications and email alerts.
5. **Accessibility in Design:** Accessibility features, such as monitoring through Android's accessibility services, have been widely adopted in recent solutions. However, these implementations often overlook ease of use. Our application simplifies the activation process, ensuring that parents can set up the system without technical expertise.

This project builds upon existing research and applications by introducing a combination of real-time monitoring, keyword detection, and app restriction alerts in a user-friendly and efficient manner. It serves as a practical, scalable, and privacy-conscious approach to addressing digital safety challenges faced by parents.

Features

The parental control application provides a robust set of features designed to ensure child safety in the digital environment. Below are the key features of the app:

1. **Keyword Monitoring**: Detects suspicious keywords in real-time from user interactions, ensuring timely alerts to parents about potential risks.
2. **Restricted App Detection**: Monitors the usage of restricted applications and immediately alerts parents if such apps are accessed.
3. **Real-Time Notifications**: Sends instant notifications to the parent's device, ensuring they are informed of any suspicious activity or restricted app usage.
4. **Email Alerts**: In addition to in-app notifications, email alerts are sent to the parent's registered email address for enhanced accessibility.
5. **User Authentication**: A secure login system ensures that only authorized parents can access and manage the app's functionalities.
6. **Multi-Language Support**: The app supports keyword detection in multiple languages, including English and Telugu, enhancing its usability for diverse users.
7. **Accessibility Features**: Leverages Android’s accessibility services to monitor text input and app usage, providing seamless integration without complex setups.
8. **Customizable Alerts**: Parents can customize the list of restricted apps and keywords to align with their specific concerns and priorities.
9. **Lightweight and Efficient**: Designed to run smoothly without impacting the device's performance, ensuring a hassle-free experience.
10. **Scalability**: The architecture is flexible, allowing the application to integrate new features or adapt to global usage scenarios in the future.

These features collectively make the parental control app a practical and effective solution for ensuring a safer digital experience for children.

Technology Stack

The development of the parental control application involves a carefully selected technology stack to ensure efficiency, scalability, and a seamless user experience. Below is an overview of the technologies used:

1. **Frontend:**
    - **Android SDK:** Utilized for building the mobile application interface and integrating device features.
    - **Kotlin Programming Language:** Ensures concise and expressive code, with enhanced safety features compared to Java.
    - **Firebase Authentication:** Provides secure login functionality for user authentication.
2. **Backend:**
    - **Spring Boot Framework:** Used to build a REST API for email notifications, ensuring a robust and scalable backend service.
    - **Maven:** Simplifies project build and dependency management for the backend.
3. **Database:**
    - **Shared Preferences (Android):** Used for lightweight data storage, such as saving user session information.
4. **Networking:**
    - **Volley Library:** Handles network operations efficiently for sending requests to the backend and receiving responses.
5. **Notifications and Alerts:**
    - **Android Notification Manager:** Sends real-time alerts to users about restricted app usage or suspicious activity.
    - **Email Notifications:** Managed via a Spring Boot API to deliver alerts to the registered email address.

6. **Accessibility Services:**
    - **Android Accessibility Service:** Monitors and intercepts user interactions to detect suspicious keywords or app usage.
7. **Development Tools:**
    - **Android Studio:** The integrated development environment (IDE) used for designing, coding, and testing the application.
    - **Postman:** Used for testing and verifying the backend email API.
8. **Testing Frameworks:**
    - **JUnit:** For unit testing the app's functionalities.
    - **Firebase Debug Tools:** For monitoring authentication and other Firebase-related integrations.

This technology stack ensures a balance between high performance, ease of development, and scalability, making it a reliable solution for the intended purpose.

System Architecture

The system architecture of the parental control application is designed to integrate various components seamlessly, ensuring real-time monitoring, secure communication, and efficient notification delivery. Below is a detailed explanation of the architecture:

1. **User Interface (Frontend):**
    - The Android application serves as the primary interface for both parents and users.
    - Provides user-friendly screens for login, managing restricted apps, and viewing notifications.
2. **Accessibility Service:**
    - Monitors user interactions on the device to detect suspicious keywords and restricted app usage.
    - Runs in the background to provide real-time alerts based on configured triggers.
3. **Notification System:**
    - Alerts the parent about suspicious activity or restricted app usage via:
        - In-app notifications using Android Notification Manager.
        - Email notifications powered by the backend service.
4. **Backend Service:**
    - **Spring Boot Framework:**
        - Handles the email notification service.
        - Processes requests from the frontend to send alerts to the registered email address.
    - **API Communication:**
        - RESTful APIs enable secure and efficient communication between the Android app and the backend server.

5. **Database and Session Management:**
    - **Shared Preferences:** Used for lightweight data storage, such as saving the logged-in user's email and session details.
6. **Networking:**
    - **Volley Library:** Facilitates sending network requests to the backend for email notifications.
7. **Security:**
    - Firebase Authentication ensures secure user login and session management.
    - HTTPS communication is implemented to secure API requests and responses.
8. **Notification Flow:**
    - When a restricted app is accessed or a suspicious keyword is detected:
        - The Accessibility Service captures the event.
        - A notification is sent in real-time via Android's Notification Manager.
        - Simultaneously, a backend API request triggers an email alert to the parent's email address.

This architecture ensures a cohesive and efficient integration of all components, providing real-time alerts and a smooth user experience. It is scalable and can accommodate additional features like advanced data analytics or cloud-based monitoring in the future.

Implementation Details

The parental control application is implemented using a combination of Android development tools, backend services, and a robust notification system. Below are the key details of the implementation:

1. **Development Framework:**
    - The application is developed using Android Studio with Kotlin as the primary programming language for Android development.
    - Backend services are built using **Spring Boot**, enabling secure email notifications.
2. **User Authentication:**
    - Firebase Authentication is integrated to allow users to log in with their email and password.
    - Shared Preferences are used to store session details, such as the logged-in email, ensuring a smooth user experience.
3. **Monitoring Service:**
    - The app employs Android's Accessibility Service to monitor device interactions.
    - The service listens for specific events such as app launches and text input changes, capturing suspicious activities in real time.
4. **Restricted App and Keyword Detection:**
    - Parents can configure a list of restricted apps and keywords through the app interface.
    - The service monitors for these apps and keywords, triggering alerts upon detection.
5. **Notification System:**
    - **In-App Notifications:** Android’s Notification Manager is used to deliver real-time alerts for detected activities.

- **Email Notifications:** Suspicious activities are reported via emails sent through a Spring Boot backend service.
- The app uses Volley to send API requests to the backend for email alerts.

6. **Backend Service:**
    - The backend is built using Spring Boot to manage email notifications.
    - It exposes RESTful APIs for secure communication with the Android app.
    - Configured to send emails to the logged-in parent's email address when notified by the app.
7. **User Interface:**
    - Features a clean, user-friendly design with screens for:
        - Login and authentication.
        - Adding or removing restricted apps and keywords.
        - Viewing notifications and alerts.
8. **System Security:**
    - HTTPS is used for secure API communication.
    - Firebase Authentication ensures the app is accessed only by authorized users.
9. **Testing and Debugging:**
    - The app has been tested on multiple Android devices to ensure compatibility and reliability.
    - Backend services were tested using Postman to validate API functionality.

This implementation structure provides a balance of simplicity, functionality, and security, making the app an effective tool for monitoring and controlling device usage.

Keyword and Restricted App Monitoring

Keyword and restricted app monitoring are the core functionalities of the parental control application, designed to help parents ensure their child's safe and appropriate use of mobile devices. Below are the details of these features:

1. **Restricted App Monitoring:**
    - Parents can add apps to a restricted list by entering the app package names through the app's user interface.
    - The application continuously monitors the foreground activities of the device using Android's **UsageStatsManager**.
    - When a restricted app is launched, the app detects this activity in real-time and triggers an alert notification for the parent.
2. **Keyword Monitoring:**
    - The application listens to text changes on the device through the **Accessibility Service API**.
    - A predefined list of suspicious keywords is stored within the app's resources, allowing parents to monitor potentially harmful or inappropriate content.
    - The service captures text input events and checks for matches with the predefined list of keywords.
    - If a keyword is detected, the app triggers a notification alert, ensuring that parents are aware of the activity.
3. **Multilingual Keyword Support:**
    - The app supports keyword detection in multiple languages, including English and Telugu.
    - Romanized Telugu words can also be detected, expanding the app's ability to cater to a diverse user base.
4. **Alerts and Notifications:**
    - Upon detecting a restricted app or keyword, the app generates:

- **In-App Alerts:** Immediate pop-up notifications to inform the parent about the activity.
- **Email Notifications:** Detailed email alerts are sent to the parent's registered email address via a Spring Boot backend service.

5. **User Control:**
    - Parents can manage the list of restricted apps and keywords dynamically through the app interface.
    - This flexibility allows them to adapt the monitoring parameters as needed.
6. **Seamless Integration:**
    - The app integrates both monitoring mechanisms into a unified system for ease of use.
    - It ensures that parents can oversee both app usage and textual activity from a single platform.

This dual approach to monitoring enhances the app's utility by addressing both app-based and text-based activities, empowering parents to protect their children in a comprehensive manner.

Fig 3: Suspicious Activity Alert

Notifications and Alerts

The **Notifications and Alerts** feature ensures parents are promptly informed about suspicious activities detected by the application. This functionality enhances the overall effectiveness of the parental control system by providing real-time updates. Below are the details:

1. **In-App Notifications:**
    - When a restricted app is opened or a suspicious keyword is detected, the app immediately generates a notification within the application.
    - These notifications include concise and clear messages to inform parents about the specific activity.
2. **Email Alerts:**
    - In addition to in-app notifications, detailed email alerts are sent to the parent's registered email address.
    - These emails contain:
        - The type of alert (e.g., restricted app usage or keyword detection).
        - The detected activity details, including the app name or keyword.
        - A timestamp for the detected event.
    - The email alerts are facilitated by a Spring Boot backend service, ensuring secure and reliable delivery.
3. **Real-Time Alerts:**
    - The app continuously monitors device activities, ensuring that alerts are sent in real-time without delays.
    - This ensures that parents can take immediate action when necessary.
4. **Customizable Notification Settings:**
    - Parents can manage their notification preferences, choosing to receive alerts only via the app, email, or both.
    - This flexibility allows parents to tailor the system to their specific needs.

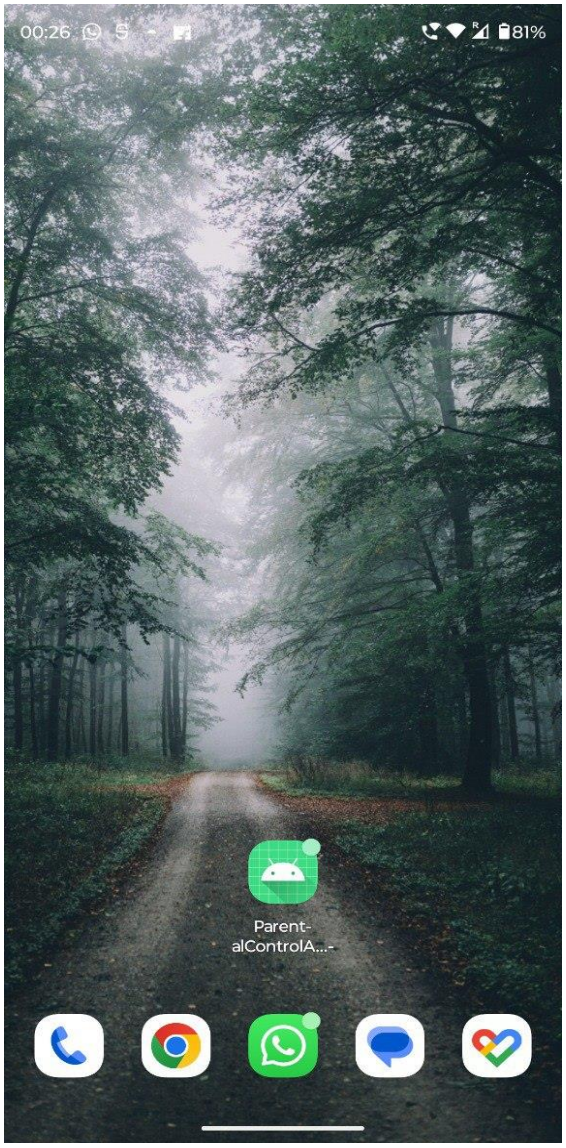
5. **Priority Handling:**
    - Alerts are prioritized based on the severity of the detected activity.
    - For instance, detecting adult content keywords or launching high-risk apps will trigger urgent alerts.
6. **Clear and Non-Intrusive Design:**
    - Notifications are designed to be informative without being intrusive, ensuring they do not disrupt the parent's workflow.
    - The system adheres to Android notification design standards, ensuring compatibility across devices.
7. **Multi-Channel Integration:**
    - Notifications are delivered seamlessly through multiple channels, such as in-app alerts and email, providing redundancy to ensure the parent receives critical updates.

By combining real-time notifications and detailed email alerts, the app provides a robust and comprehensive alerting system. This ensures that parents are always informed about potentially harmful activities, allowing them to take timely action to safeguard their children.

Fig 4: Alert Example

Screenshots with Explanation

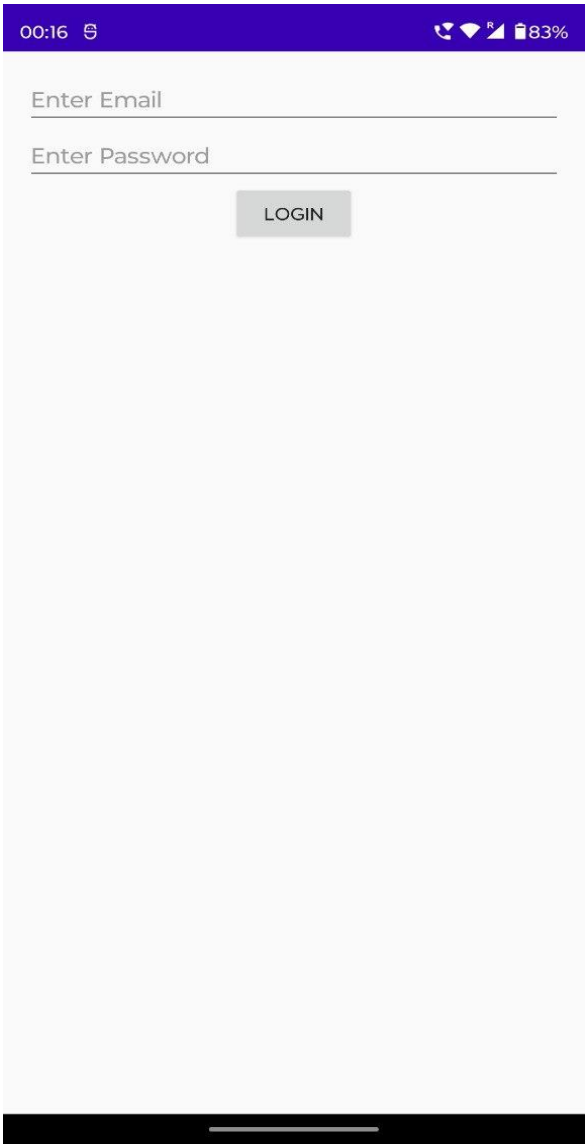
Screenshot 1: **App Icon on Home Screen**



Explanation:

This screenshot highlights the placement of the Parental Control App icon on the home screen of the Android device. The icon represents the application's visual identity and demonstrates its seamless integration into the Android ecosystem, ensuring accessibility and ease of use for parents.

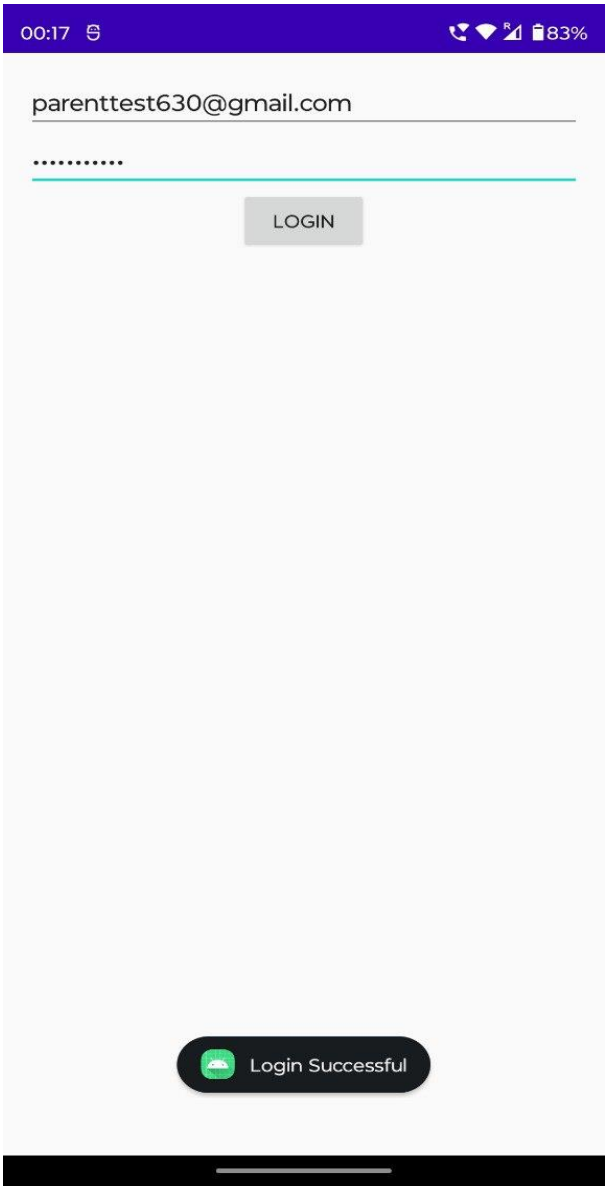
Screenshot 2: **Login Page**



Explanation:

This screenshot displays the secure login page of the application, where parents can enter their email and password to access the app's features. It highlights the importance of authentication to ensure data security and restricted access.

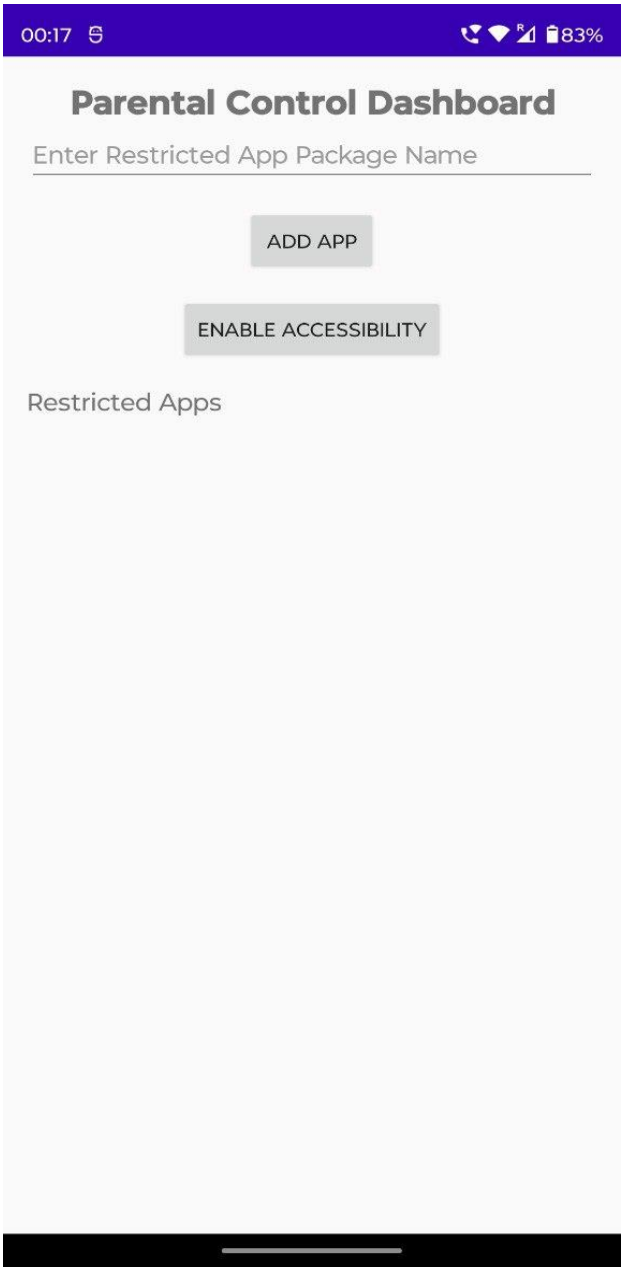
Screenshot 3: **Login Successful Notification**



Explanation:

This screenshot confirms successful login with a notification at the bottom of the screen. It demonstrates a smooth authentication process, ensuring only authorized users can manage the app's functionalities**.**

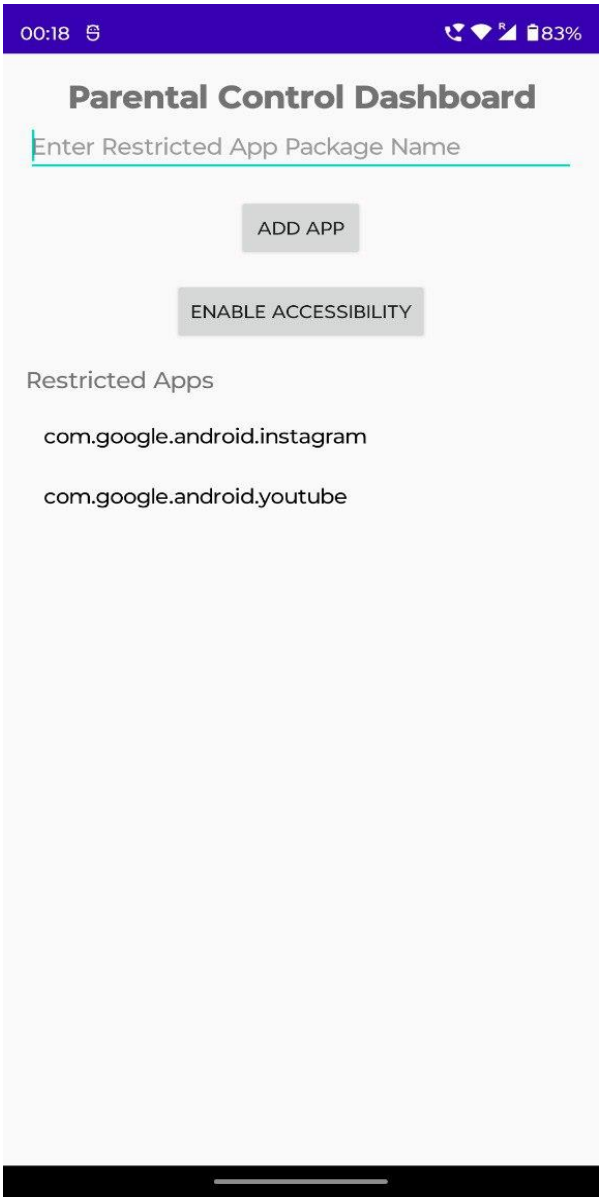
Screenshot 4: **Parental Control Dashboard**



Explanation:

This screenshot illustrates the dashboard's user interface, where parents can input the package name of apps to be restricted. The "Add App" and "Enable Accessibility" buttons enable parents to manage app restrictions conveniently.

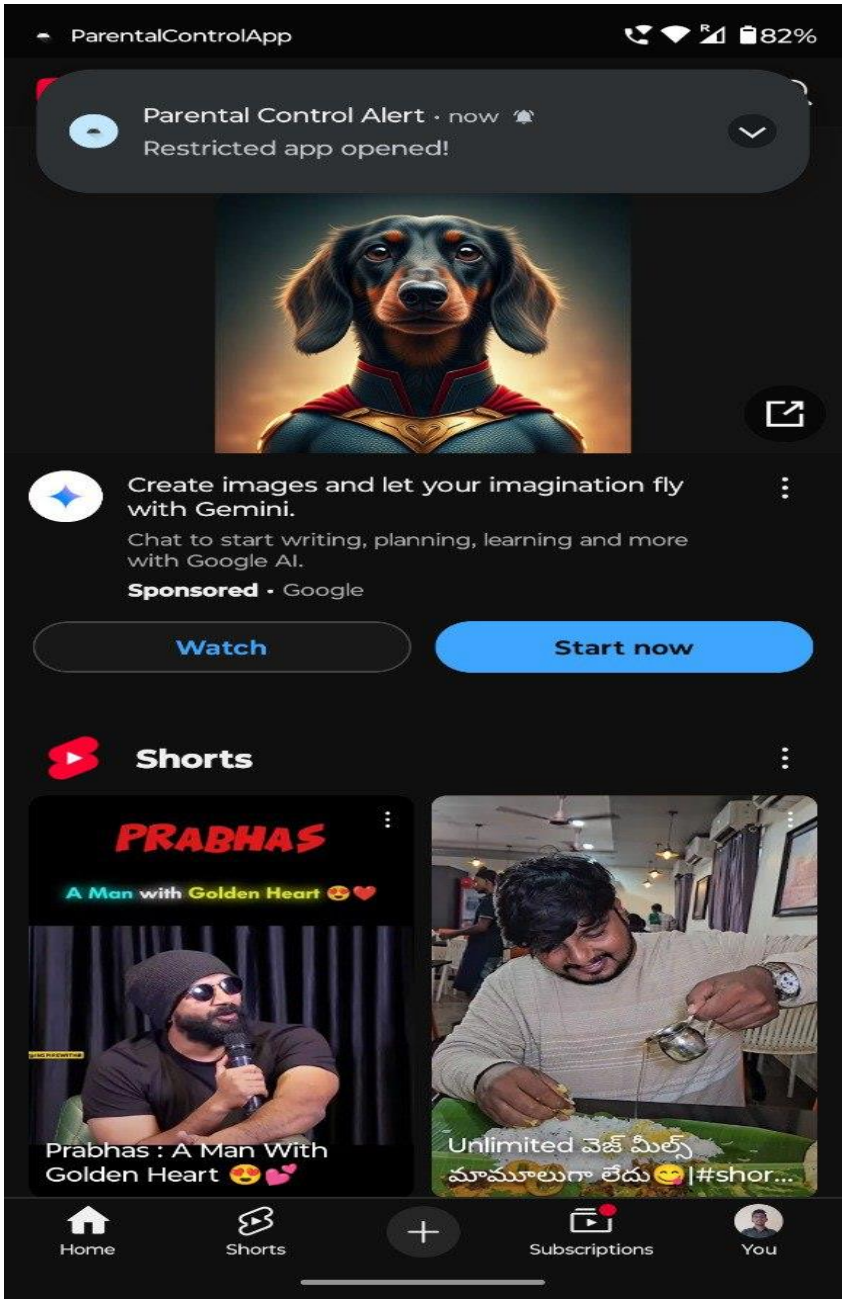
Screenshot 5: **Parental Control Dashboard with Restricted Apps**



Explanation:

This screenshot shows a list of restricted apps successfully added to the dashboard. It emphasizes the ease of managing restrictions, allowing parents to monitor and control app usage effectively.

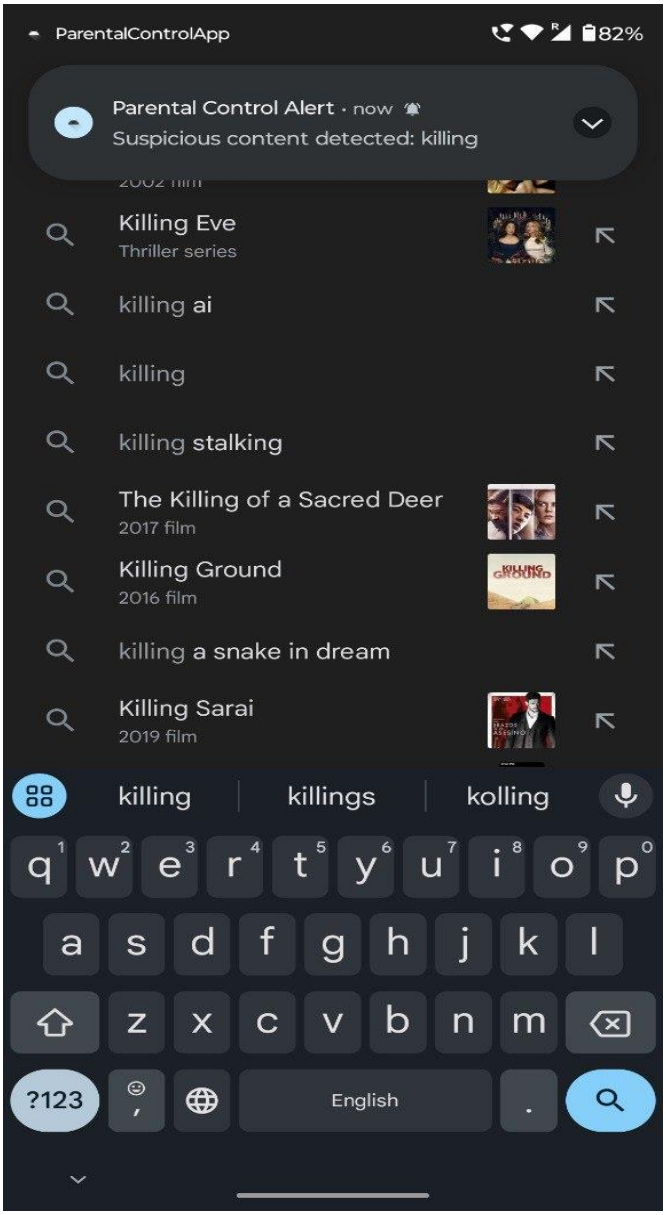
Screenshot 6: **Restricted App Alert**



Explanation:

This screenshot showcases the notification triggered when a restricted app, like YouTube, is opened. The app continuously monitors and restricts unauthorized app usage, sending a real-time alert to the parent.

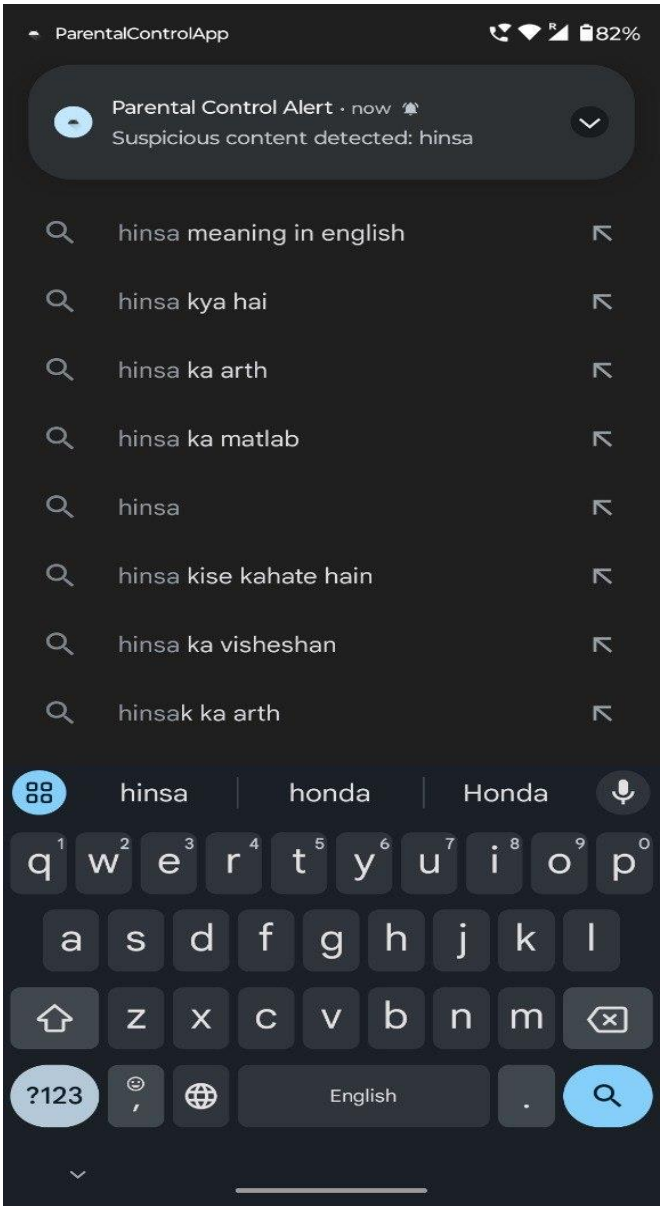
Screenshot 7: **Notification for Suspicious Keyword "Killing"**



Explanation:

This screenshot highlights the app's notification system in action when a sensitive keyword, such as "killing," is detected. The app instantly generates an alert at the top of the screen to inform the parent.

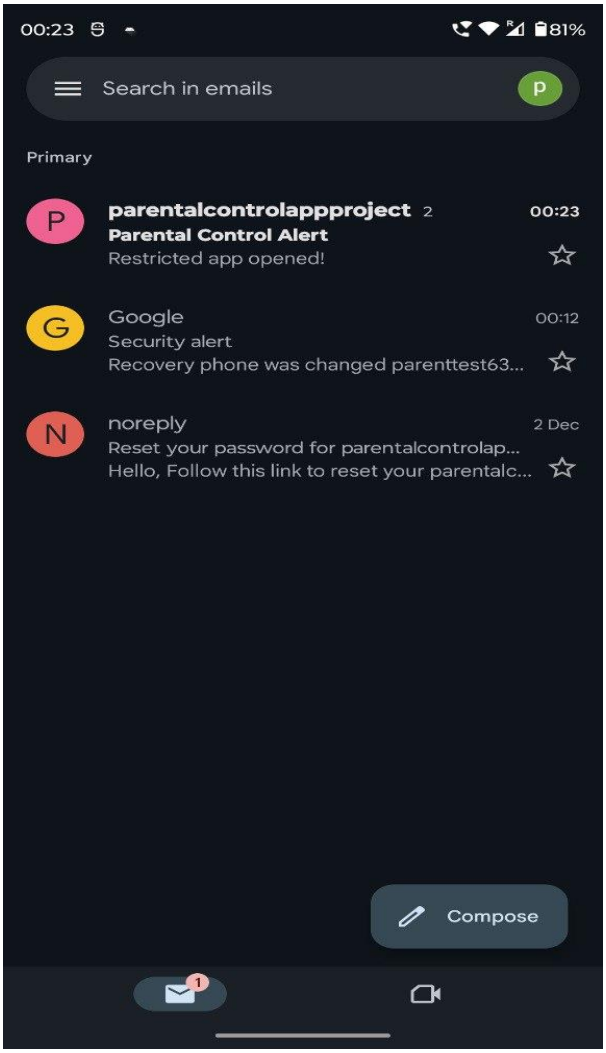
Screenshot 8: **Search with Suspicious Telugu Keyword Alert**



Explanation:

This screenshot depicts the app identifying and sending an alert for a suspicious Telugu keyword ("hinsa") entered in the search bar. The notification displayed at the top confirms the app's proactive monitoring feature.

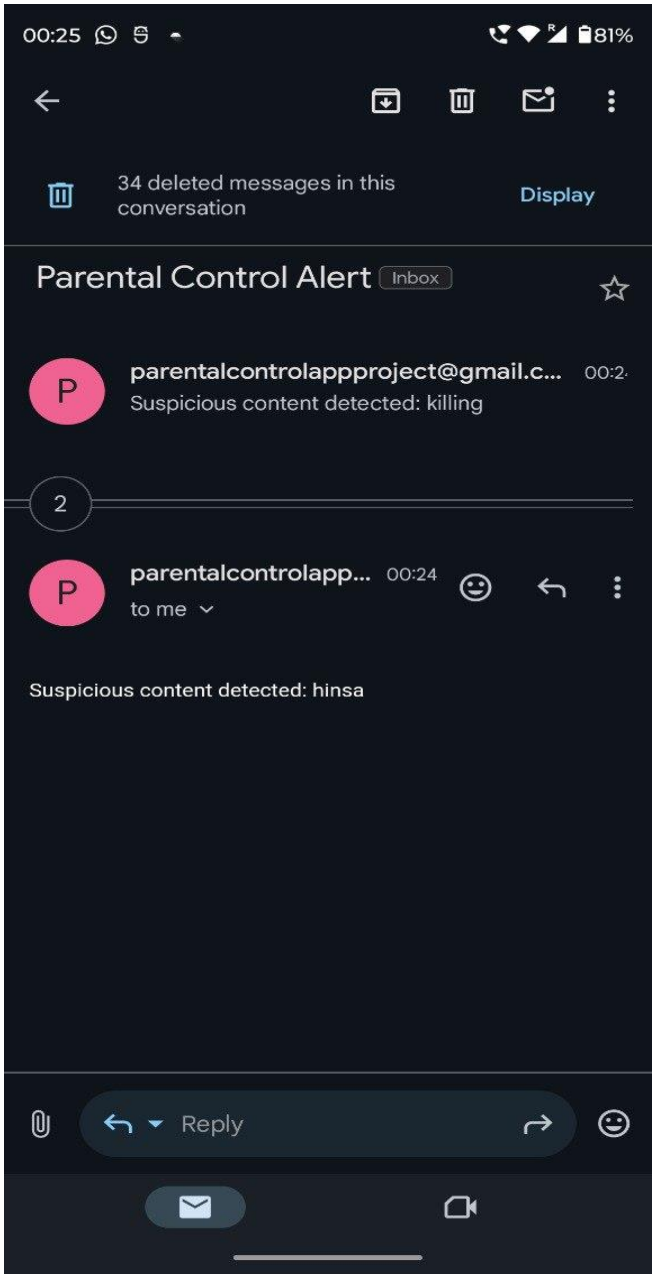
Screenshot 9: **Email List Showing Notifications**



Explanation:

This screenshot shows a consolidated view of email alerts in the parent's inbox. Each notification provides details about detected suspicious activity, such as opening a restricted app or identifying inappropriate text content. This ensures the parent can stay updated on their child's activities in real-time.

Screenshot 10: **Email Notification with Detected Suspicious Keywords**



Explanation:

This screenshot displays the email notification received by the parent when the app detects a suspicious keyword, such as "killing" or "hinsa." It demonstrates the app's capability to analyze text content and promptly notify parents about potentially harmful activities.

Use Cases

1. **Child's Online Safety**:
   Parents can monitor and restrict access to inappropriate content, ensuring a safe online experience for their children.

2. **Proactive Notifications**:
   Real-time alerts for suspicious keywords or restricted app usage provide timely information for parental intervention.

3. **Customizable Restrictions**:
   Parents can customize app restrictions to suit the specific needs of their child, such as blocking social media or gaming apps during study hours.

4. **Multi-Device Monitoring**:
   The application supports monitoring across multiple devices, enabling broader supervision for families with multiple children.

5. **Language-Specific Monitoring**:
   The app's ability to detect suspicious activity in multiple languages, including Telugu and English, ensures wider applicability.

6. **Educational Supervision**:
   Parents can block entertainment apps during school hours or allow only educational apps, promoting a focused learning environment.

7. **Activity Insights**:
   The application can log and report usage patterns, helping parents analyze their child's activities over time.

8. **Emergency Notifications**:
   If the app detects sensitive or dangerous activities, such as keywords related to violence or self-harm, parents receive immediate alerts.

9. **Easy-to-Use Dashboard**:
   The intuitive dashboard makes it easy for parents to manage app restrictions, notifications, and other settings without technical expertise.

10. **Improved Family Communication**:

    By fostering transparency between parents and children regarding device usage, the app helps build trust and promotes responsible behaviour.

Fig 5: Need for Monitoring

Conclusion

The **Parental Control Application** is designed to empower parents in ensuring a safe and productive digital environment for their children. With features such as keyword and restricted app monitoring, real-time notifications, and email alerts, the app provides an effective way to supervise and guide children’s device usage. Its customizable and language-supportive capabilities make it versatile for families with diverse needs.

This project demonstrates how technology can be leveraged to address modern parenting challenges, offering a proactive solution to monitor and manage children's online activities. By enabling parents to take informed actions and fostering responsible digital behavior, the app contributes to creating a safer and healthier online ecosystem.

Through this project, we have gained valuable insights into application development, accessibility services, and integrating backend services, which further highlight the practical implementation of technical knowledge to solve real-world problems.

References

- Firebase Documentation - For implementing authentication and backend services. Firebase Official Documentation
- Android Developers Documentation - For understanding Android accessibility services and system-level features. Android Official Documentation
- Udemy - Online courses for Android development concepts and project structure.
- YouTube Tutorials - For practical demonstrations and troubleshooting during application development.
- Stack Overflow - Community-driven solutions to resolve coding challenges and implementation issues. Stack Overflow
- Official Kotlin Documentation - For efficient usage of Kotlin programming language. Kotlin Documentation
- Volley Library Documentation - For integrating API calls and network communication. Volley Official Documentation
- Research Papers and Articles on Parental Control Apps - For understanding the importance and scope of parental control applications.
- Practical insights and guidance from faculty members during the development process.