

Final Year Project Ideas for Cross-Platform Development

Build Apps for Android & iOS Using a Single Codebase

Website Name: haas.dev

Website Link: <https://dev-roast-app.vercel.app>

Introduction

Cross-platform development allows you to **build apps for multiple platforms with one codebase**.

This guide provides **final year project ideas** using frameworks like React Native and Flutter.

1. Personal Productivity Apps

- **To-Do List App** – Sync tasks across devices
 - **Notes App** – Store text, images, and audio with local and cloud storage
 - **Habit Tracker** – Daily habit reminders with analytics
-

2. Social & Communication Apps

- **Chat Application** – Real-time messaging with Firebase or Socket.io
 - **Event Planner App** – Invite friends, RSVP, notifications
 - **Social Feed App** – Post updates, images, likes, and comments
-

3. E-Commerce & Business Apps

- **Shopping App** – Product catalog, cart, checkout, payment integration
 - **Inventory Management App** – Track stock for small businesses
 - **Expense Tracker** – Categorize expenses, visualize data with charts
-

4. Learning & Educational Apps

- **Quiz App** – Timed quizzes with scoring and leaderboard
- **Flashcards App** – Study tool with spaced repetition

- **Language Learning App** – Vocabulary practice with audio and quizzes
-

5. Libraries & Tools You Can Use

- **UI & Animations:** React Native Paper, Flutter Material, Lottie
 - **Networking:** Axios (React Native), Dio (Flutter)
 - **Storage:** AsyncStorage, Realm, SQLite
 - **State Management:** Redux, Context API, Provider, Riverpod
 - **Notifications & Background Tasks:** Firebase Cloud Messaging, WorkManager (Flutter)
-

6. Mini Project Ideas to Apply Skills

1. **Fitness Tracker App** – Track workouts and calories
 2. **News Aggregator App** – Fetch news from multiple APIs
 3. **Chat App** – Real-time communication with notifications
 4. **Expense Tracker** – Multi-platform app with charts and offline storage
-

7. Best Practices

- Use **modular architecture** (MVVM or Clean Architecture)
 - Test on **both Android and iOS devices**
 - Optimize network calls and performance for mobile
 - Use **Git** for version control and collaboration
-

8. Key Takeaways

- Cross-platform development saves time by **writing once, running anywhere**
 - Focus on projects that are **portfolio-ready and real-world practical**
 - Integrate libraries and tools to simplify development and enhance functionality
 - Hands-on projects improve employability and developer skills
-

