International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 03 | March - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

Books on Rent Application

Prof Nikita N.Jain,Abhishek Londhe,Vedant Kudale,Sanket Lonkar,Sourabhkumar Singh

B.E Computer Department ,JSPM Narhe Technical Campus

Abstract - The increasing demand for affordable and accessible reading resources has led to the development of digital solutions in the book rental domain. This paper presents a Books on Rent Application using Flutter, a cross-platform mobile application designed to simplify and automate the book rental process. The system allows users to browse, rent, and return books efficiently through a user-friendly interface. The application is developed using Flutter and Dart, ensuring smooth performance across Android and iOS platforms. Firebase is used as the backend for real-time database management, authentication, and cloud storage, while Razorpay is integrated for secure payment processing. The system includes modules such as user management, book catalog, rental management, payment processing, and notifications. The application improves accessibility, reduces cost, and promotes sustainable reading habits. Performance evaluation shows high reliability, fast response time, and scalability. The proposed system effectively bridges the gap between book providers and readers through a digital platform.

Key Words: Flutter, Book Rental System, Firebase, Mobile Application, Razorpay, Cloud Database

1.INTRODUCTION

In the digital era, access to books has evolved from traditional libraries to online platforms. However, purchasing books can be expensive and inconvenient for short-term use. The Books on Rent Application provides a cost-effective and efficient solution by enabling users to rent books digitally.

The system is developed using Flutter, allowing cross-platform compatibility, and integrates Firebase for backend services. Users can browse books, rent them, and make secure payments through an intuitive interface.

The platform serves both users and administrators. Users can search, rent, and manage books, while admins can manage inventory and transactions. This system enhances accessibility, affordability, and promotes digital transformation in the book rental domain

2. LITERATURE SURVEY

Recent studies highlight the growing adoption of mobile-based book rental and e-commerce systems to improve accessibility and affordability of reading resources. Many researchers have explored the use of cross-platform frameworks like Flutter for building scalable and user-friendly applications. Flutter enables faster development and consistent UI across Android and iOS platforms.

Several works also emphasize the role of cloud-based backend services such as Firebase in providing real-time database management, authentication, and storage. These systems improve performance and scalability while reducing development complexity.

Research on digital payment integration shows that payment gateways like Razorpay and Paytm enhance user trust and ensure secure transactions in mobile applications. However, many existing systems lack proper rental management features, real-time notifications, and offline accessibility.

This project improves upon existing solutions by integrating Flutter, Firebase, and Razorpay into a single platform that provides a complete and efficient book rental system.

3.PROBLEM STATEMENT

Traditional book rental systems suffer from:

- Manual processes and physical dependency
- Lack of proper tracking system
- No real-time updates or notifications
- Limited availability of books

Existing online systems mainly focus on book selling rather than renting and often lack flexibility. This creates a need for a scalable and automated digital rental system.

4.OBJECTIVE

- To develop a mobile-based book rental application using Flutter and Dart.
- To implement secure user authentication and management.
- To integrate online payment functionality using Razorpay.
- To provide real-time data management using Firebase.
- To enable notifications for due dates and updates.
- To promote affordable and sustainable access to books.

5. SYSTEM ARCHITECTURE

The proposed system follows a modular, layered architecture that separates presentation, business logic, and data management to ensure scalability, maintainability, and high performance.

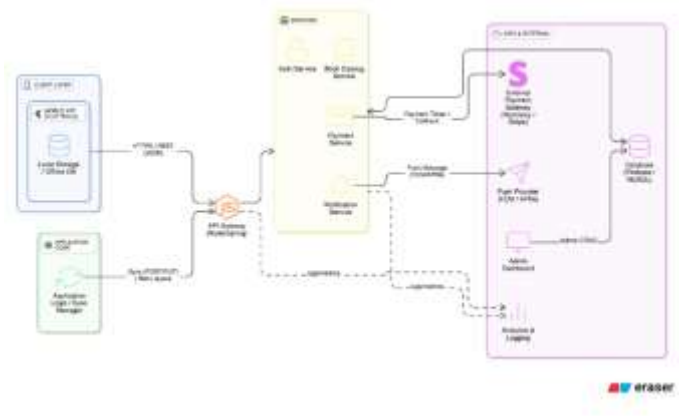

Figure 1: System Schematic (Block Diagram Overview)

The architecture is divided into four primary layers:

1. **Client Layer (Frontend)**
- Built using Flutter framework and Dart language.
- Provides user interface components such as dashboards, book listings, search filters, and payment screens.
- Supports cross-platform execution on Android and iOS with a single codebase.
- Implements local caching (SQLite/Hive) for limited offline access.

2. **Application Layer (Business Logic)**
- Handles core functionalities such as authentication, book rental logic, payment validation, and notifications.
- Uses state management techniques (Provider / Riverpod / Bloc) to manage UI and data flow efficiently.

3. **Backend Layer (Cloud Services)**
- Firebase Firestore: Real-time NoSQL database for storing books, users, and rental data.
- Firebase Authentication: Secure login using email/password or Google Sign-In.
- Firebase Storage: Stores book images and media files.
- Firebase Cloud Messaging (FCM): Sends real-time notifications.

4. **External Services Layer**
- Razorpay Payment Gateway: Handles secure online transactions.
- Optional Backend API (Node.js / Spring Boot): Used for advanced logic, analytics, and payment verification via webhooks.

Data Flow Architecture

1. User interacts with the Flutter app.
2. Requests are sent to Firebase or backend APIs.
3. Backend processes authentication, payment, and data updates.
4. Firestore updates real-time database.
5. Notifications are sent via FCM.

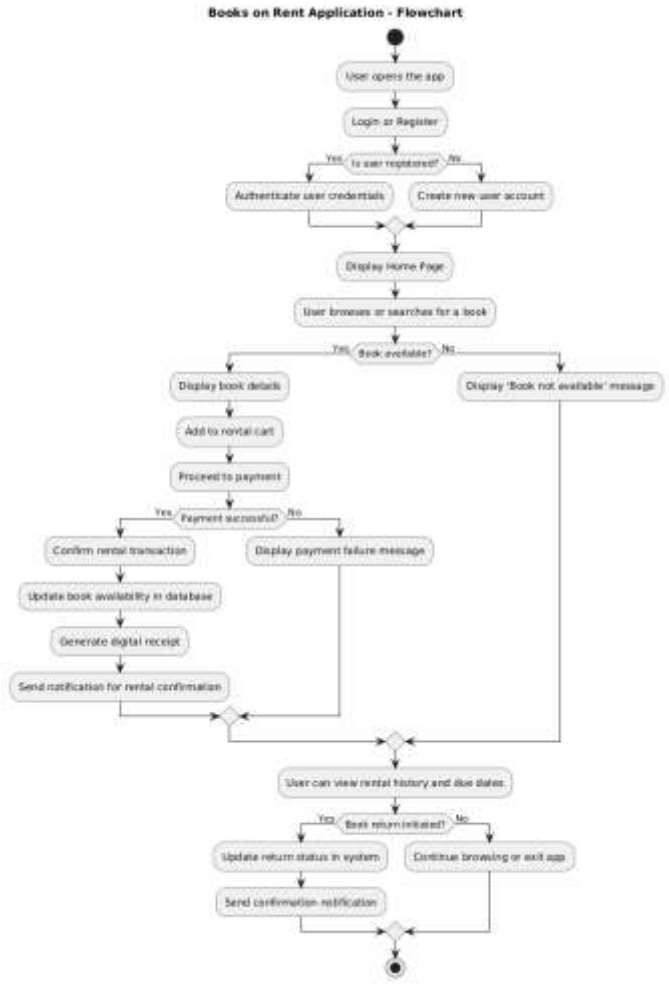

Figure 2: Data Flow Diagram(Flowcahart)

6. METHODOLOGY

5.1 User Module
- User registration and login
- Profile management
- Rental history tracking

5.2 Book Management Module
- Add, update, and delete books
- Categorization and search functionality

5.3 Rental Module
- Renting and returning books
- Availability tracking

5.4 Payment Module
- Secure payment processing using Razorpay
- Transaction verification

5.5 Notification Module
- Alerts for due dates and confirmations
- Admin notifications

7. IMPLEMENTATION

The application is implemented using modern technologies:

- Flutter and Dart for frontend development
- Firebase Firestore for database management
- Firebase Authentication for login system
- Razorpay API for payment processing
- Firebase Cloud Messaging for notifications

8. RESULTS AND DISCUSSION

The developed Books on Rent Application was tested under various conditions to evaluate its performance, usability, and reliability. The results indicate that the system performs efficiently and provides a smooth user experience.

8.1 Performance Analysis

Parameter	Result Observed	Description
Response Time	< 2 seconds	Fast loading of book data and screens
Payment Processing	3–5 seconds	Secure and quick transaction via Razorpay
Database Retrieval	Real-time	Instant data updates using Firebase
App Stability	High	No crashes during testing

8.2 Comparison with Traditional System

Feature	Traditional System	Proposed System
Accessibility	Limited	Available anytime, anywhere
Payment Mode	Cash only	Online secure payments
Tracking System	Manual	Automated tracking
Notifications	Not available	Real-time alerts
Efficiency	Low	High

8.3 Discussion

The results demonstrate that the proposed system significantly improves efficiency, accessibility, and user experience compared to traditional book rental systems. The integration of Flutter and Firebase ensures real-time performance, while Razorpay enhances secure payment handling. Overall, the application meets its objectives and provides a scalable and reliable solution.

9. CONCLUSION

The Books on Rent Application using Flutter provides an efficient and modern solution to overcome the limitations of traditional book rental systems. By leveraging cross-platform mobile development with Flutter and integrating cloud-based backend services such as Firebase, the system ensures real-time data management, scalability, and a seamless user experience.The application successfully automates key processes including book browsing, renting, returning, and payment handling. The integration of Razorpay enables secure and reliable online transactions, while Firebase Cloud Messaging ensures timely notifications for users regarding due dates and updates. This significantly enhances user convenience and reduces manual effort.

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to our project guide for their valuable guidance, continuous support, and encouragement throughout the development of this project. Their insights and suggestions helped us in successfully completing this work.We are also thankful to the faculty members of the Department of Computer Engineering for providing the necessary resources and knowledge required for this project. Their constant motivation played a crucial role in enhancing our understanding.

REFERENCES

1. Google, "Firebase Documentation," Available: https://firebase.google.com/docs

2. Google, "Flutter Documentation," Available: https://flutter.dev/docs

3. Razorpay, "Razorpay Payment Gateway API Documentation," Available: https://razorpay.com/docs

4. Dart Team, "Dart Programming Language Guide," Available: https://dart.dev/guides

5. Pub.dev, "Flutter Packages Repository," Available: https://pub.dev

6. Kumar, S., & Patel, D., "Design and Development of Flutter-based E-commerce Applications," *International Journal of Emerging Technologies in Computer Science*, 2023.

7. Singh, R., & Joshi, A., "Cloud-Based Mobile Applications Using Firebase for Real-Time Data Management," *IJERT Journal of Engineering Research*, 2024.

8. AWS, "Amazon S3 Cloud Storage Documentation," Available: https://aws.amazon.com/s3

9. Google Cloud, "Cloud Firestore Documentation," Available: https://cloud.google.com/firestore/docs

10.Android Developers, "Android Studio Development Guide," Available: https://developer.android.com/studio

11.Provider Package, "State Management in Flutter," Available: https://pub.dev/packages/provider

12.Pressman, R. S., "Software Engineering: A Practitioner's Approach," 8th Edition, McGraw-Hill, 2015.