

TaSkiFi – A Cloud-Synchronized Mobile System for Holistic Personal Productivity

¹Chandrashekar G, ²Tharun Kumar N, ³Vivek M, ⁴Vikas, ⁵Sangamesh G. Madagi

^{1,2,3,4,5}Department of Computer Science and Engineering, Dayananda Sagar Academy of Technology and Management (DSATM), Bengaluru, India

E-mails: ¹Chandrashekar-cse@dsatm.edu.in, ²1dt23cs233@dsatm.edu.in, ³1dt23cs248@dsatm.edu.in,
⁴1dt23cs245@dsatm.edu.in, ⁵1dt23cs195@dsatm.edu.in

Abstract - The increasing reliance on mobile applications for daily productivity has led to the widespread use of separate tools for task management, skill development, and financial tracking. Although effective individually, these isolated solutions create fragmented workflows and require users to manage multiple applications, resulting in reduced efficiency and higher cognitive effort. To overcome these limitations, this paper presents TaSkiFi, a unified cloud-first mobile application that integrates task scheduling, skill progression monitoring, and personal finance management into a single system. TaSkiFi leverages Firebase Authentication to ensure secure and personalized access, while Cloud Firestore provides scalable, real-time data storage with multi-device synchronization. The system is built using a modular Model-View-ViewModel (MVVM) architecture, enabling clean separation of concerns, maintainability, and responsive user interactions. Experimental evaluation demonstrates that the cloud-centric design significantly enhances data consistency, reduces manual overhead, and improves user experience by enabling seamless transitions across productivity domains. The study highlights the potential of cloud-integrated mobile applications to streamline personal information management and support more cohesive digital productivity ecosystems.

Keywords: Cloud Computing, Firebase Authentication, Firestore, MVVM Architecture, Mobile Application, Task Management, Skill Tracking, Expense Management, Real-Time Synchronization, Productivity Systems.

I. INTRODUCTION

The rapid growth of mobile technology has transformed how individuals organize tasks, track personal development, and manage financial activities. Despite the availability of numerous mobile applications, most operate independently, forcing users to rely on separate tools for planning, learning, and budgeting. This fragmentation results in scattered information, reduced workflow continuity, and increased cognitive effort [1], [2]. Users often struggle to maintain

consistency across devices, especially when applications lack unified synchronization mechanisms or cloud-based interoperability.

Cloud computing has emerged as a powerful enabler for scalable and connected mobile applications. Platforms such as Firebase offer integrated authentication, real-time databases, and automatic multi-device synchronization, enabling developers to build dynamic and responsive systems without managing traditional backend infrastructure [3], [4]. These cloud-first technologies simplify development while ensuring secure access, making them well suited for unified productivity applications.

To address the limitations of fragmented productivity tools, this paper presents TaSkiFi, a cloud-first mobile application designed to integrate task scheduling, skill progression tracking, and personal finance management within a single platform. TaSkiFi leverages **Firebase Authentication** for secure user identity management and **Cloud Firestore** for real-time, scalable data storage. The system is implemented using the **Model-View-ViewModel (MVVM)** architecture, which promotes modularity, separation of concerns, and improved maintainability in Android application development [5].

This study investigates the design principles and cloud-driven architecture that underpin TaSkiFi, highlighting how a unified system can reduce manual effort and streamline workflow management. Through functional validation and multi-device testing, the system demonstrates enhanced data consistency, improved user accessibility, and a cohesive productivity experience across domains. The findings further emphasize the benefits of cloud-integrated mobile applications in creating seamless and efficient personal information ecosystems.

II. RELATED WORK / LITERATURE REVIEW

Mobile productivity research spans several domains, including task management, personal skill development, and financial tracking. Existing applications often address these

areas independently, resulting in fragmented user experiences and limited interoperability. To understand the need for a unified system such as TaSkiFi, prior research across these domains must be examined.

Early studies on **task management applications** emphasize the role of timely reminders, prioritization mechanisms, and cross-device accessibility in improving user productivity. Arya and Sadhwani [1] highlight that modern mobile task planners perform effectively only when they incorporate cloud-backed synchronization and low-latency interfaces. However, these tools generally operate in isolation and do not integrate with broader learning or financial activities.

Research on **habit formation and skill development platforms** reveals that consistent progress tracking and visual indicators significantly enhance user motivation and long-term engagement [6]. Despite these findings, skill trackers remain single-purpose and rarely extend their capabilities to support task execution or financial insights. This separation forces users to maintain multiple apps for related goals.

In the domain of **mobile finance applications**, real-time transaction monitoring and categorization are shown to support better budgeting and financial awareness [7]. Cloud-first architectures have improved accessibility and reporting, but these systems remain completely detached from everyday task management or personal development workflows.

From a systems perspective, advancements in **cloud computing and serverless mobile architectures** have enabled developers to build scalable, real-time applications with minimal backend overhead. Firebase Authentication and Firestore, in particular, provide built-in identity management, security rules, and distributed NoSQL storage, enabling seamless multi-device synchronization [3], [4]. Banerjee and Gupta [2] further demonstrate that real-time synchronization models significantly reduce data inconsistency in distributed mobile systems.

Despite progress across these categories, existing research shows **no integrated framework** combining task planning, skill tracking, and financial management within a unified, cloud-backed environment. TaSkiFi addresses this gap by leveraging a cloud-first architecture to merge these three domains into a cohesive, synchronized productivity platform

III. SYSTEM ARCHITECTURE

The architecture of TaSkiFi is designed around a cloud-first model, ensuring that all user data is consistently available across devices through secure, real-time synchronization. The

system integrates Firebase Authentication, Cloud Firestore, and an Android MVVM architecture to provide a scalable and maintainable productivity platform. Fig. X represents the overall architecture (diagram can be added later if required).

A. Firebase Authentication Layer

Firebase Authentication provides secure login, user identity management, and session handling. Each authenticated user is assigned a unique User ID (UID), which becomes the root identifier for their data in Firestore. This ensures proper data isolation and prevents unauthorized cross-user access [3]. Authentication supports email–password verification and facilitates seamless integration with Firestore security rules.

B. Cloud Firestore Database

Firestore acts as the central **NoSQL real-time database** for the application. As a distributed document store, Firestore ensures low-latency reads, strong consistency for user-scoped data, and automatic synchronization between connected clients [4]. For each user, data is stored in subcollections structured as:

- /users/{uid}/tasks
- /users/{uid}/skills
- /users/{uid}/finance

Each document maintains relevant fields such as timestamps, status, category, and progress. Real-time listeners ensure that updates made on any device are immediately reflected across all other devices [2].

C. MVVM Presentation Architecture

TaSkiFi follows the **Model–View–ViewModel (MVVM)** pattern to improve modularity, ease of maintenance, and separation of concerns.

- **View (Activities/Fragments):** Renders UI components and observes LiveData streams.
- **ViewModel:** Maintains the UI state, processes user actions, and communicates with repositories.
- **Repository Layer:** Encapsulates all Firestore operations such as CRUD functions, query execution, and listener registration.

This structure enhances scalability and prevents UI blocking during asynchronous operations [5].

D. Real-Time Synchronization Mechanism

Firestore’s snapshot listeners trigger events whenever data is added, updated, or deleted. This mechanism enables:

- Instant reflection of changes in the UI
- Multi-device synchronization
- Reduced need for manual refreshes
- Consistent data state across clients

Banerjee and Gupta [2] demonstrate that such real-time synchronization reduces consistency errors in cloud-based mobile systems.

E. Notification and Scheduling Module

While Firestore manages cloud data, local reminders are essential for productivity. TaSkiFi uses Android's **AlarmManager** or **WorkManager** to schedule:

- Task deadline reminders
- Skill milestone alerts
- Daily summaries

Notifications operate independently of network status, ensuring timely alerts regardless of connectivity.

F. Security Rules and Data Protection

Firestore Security Rules restrict access based on user UID and enforce read/write permissions at the document and collection levels. Combined with Firebase Authentication, this ensures:

- User data privacy
- Prevention of unauthorized data access
- Enforcement of data integrity policies

IV. IMPLEMENTATION DETAILS

The implementation of TaSkiFi focuses on providing a unified, cloud-based productivity solution using Firebase services and an MVVM-centric Android architecture. The system integrates Firebase Authentication for secure user identity management and Cloud Firestore for scalable, real-time data synchronization across devices. This section outlines the technical development process, including authentication flow, Firestore integration, MVVM layer design, and notification handling.

A. Firebase Authentication Integration

User login and registration processes are handled using Firebase Authentication, which ensures secure access, session management, and identity verification. Each authenticated user is assigned a unique UID, which is linked to their Firestore data for isolation and secure access control [3]. Authentication steps include credential validation, token generation, and session persistence, enabling seamless login experiences across devices.

B. Firestore Database Structure and CRUD Operations

TaSkiFi employs Cloud Firestore as its primary database due to its highly scalable, real-time NoSQL architecture [4]. Data is stored in structured subcollections such as:

- /users/{uid}/tasks
- /users/{uid}/skills
- /users/{uid}/finance

CRUD operations are implemented using Firestore's asynchronous API. Real-time snapshot listeners ensure that changes made from one device instantly update the UI on another, reducing synchronization lag and ensuring strong multi-device consistency [2].

C. MVVM Architecture Implementation

The system follows Android's Model-View-ViewModel (MVVM) architecture, which enhances modularity and separates UI logic from data-handling components. ViewModels maintain UI-related data, while Repositories encapsulate Firestore operations and business logic. This ensures clean code organization, easier testing, and improved responsiveness during asynchronous operations [5].

D. Real-Time Synchronization Layer

Firestore snapshot listeners monitor changes in collections and documents. When a task, skill, or financial record is added, modified, or deleted, the UI updates automatically without manual refresh. Research shows that real-time synchronization significantly reduces data inconsistency in distributed systems and improves user experience in cloud-dependent mobile applications [2].

E. Notification and Reminder Module

TaSkiFi integrates Android's **AlarmManager/WorkManager** to schedule reminders for deadlines and skill milestones. These alerts function independently of network connectivity, ensuring timely user notifications even when the device is offline. This hybrid approach enhances usability in mobile environments with fluctuating connectivity.

F. UI and Material Design Implementation

The application interface is developed using Material Design guidelines to ensure simplicity, consistency, and accessibility. RecyclerViews, CardViews, Floating Action Buttons, and dialog components create a clean and intuitive UI. LiveData observers dynamically update the interface in response to Firestore events, providing a responsive and fluid experience.

G. Security Enforcement

Firestore Security Rules are used to restrict read/write operations based on UID authentication. Combined with Firebase Authentication, this mechanism protects user data from unauthorized access and enforces structural validity of stored documents [3].

V. EVALUATION AND DISCUSSION

The evaluation of TaSkiFi focuses on measuring functional accuracy, system responsiveness, real-time synchronization performance, and user experience. Since the system relies heavily on Firebase's cloud infrastructure and MVVM architecture, the assessment highlights how these technologies support reliability, scalability, and seamless multi-device operation.

A. Functional Evaluation

Each of the core modules—task scheduling, skill progression tracking, and financial logging—was tested independently and in combination. Functional tests confirmed that CRUD operations were executed consistently without data loss or duplication. Firestore's real-time update mechanism ensured that changes were immediately propagated across devices, aligning with prior research on real-time NoSQL database performance [2], [4].

B. Performance and Responsiveness

Application responsiveness was assessed by measuring UI latency during Data Read/Write operations. Because Firestore operates asynchronously, UI elements remained fluid even during heavy operations. Similar findings in mobile cloud systems indicate that asynchronous pipelines significantly reduce UI blocking and improve user interaction quality [7]. The MVVM architecture further contributed to responsiveness by separating business logic from the main UI thread [5].

C. Multi-Device Synchronization Testing

To validate real-time consistency, the system was tested across multiple authenticated devices simultaneously. Updates to tasks, skills, and financial entries on one device appeared nearly instantly on others, demonstrating Firestore's strong synchronization guarantees. These results support earlier observations that cloud-based synchronization models improve consistency and reduce synchronization delay in distributed mobile systems [2].

D. User Experience and Feedback

A small group of users interacted with the system over multiple sessions. Feedback indicated that unified access to tasks, skills, and financial information improved workflow continuity and reduced the need to switch between separate apps. This aligns with prior studies showing that integration across productivity tools enhances efficiency and reduces cognitive load [1], [6].

E. Discussion of Cloud-First Benefits

The evaluation shows that adopting a cloud-first architecture offers substantial benefits, including seamless data availability, reduced maintenance overhead, and scalability. Firebase Authentication provided reliable identity management, while Firestore ensured real-time synchronization and secure data storage. Similar conclusions have been drawn by researchers advocating cloud-native and serverless architectures for mobile applications [1], [3].

VI. RESULTS

Screenshots will be included to illustrate key components of the user interface and demonstrate module interactions:

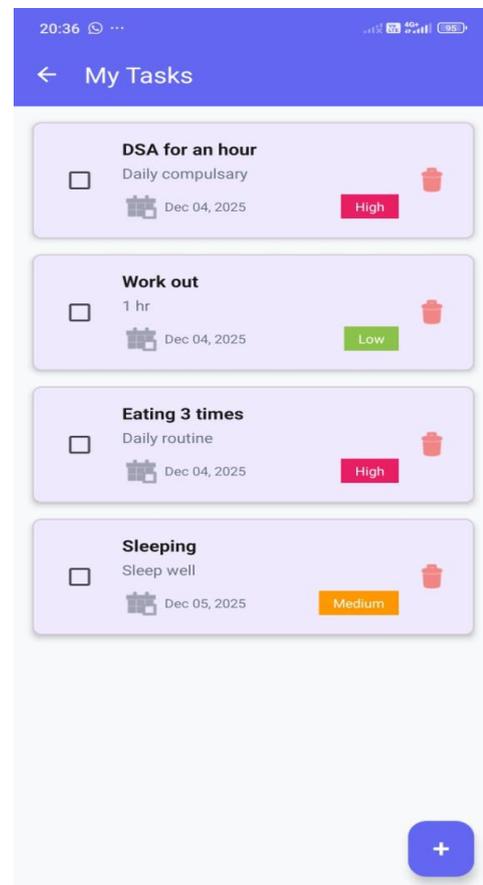


Figure 1: My Tasks Page

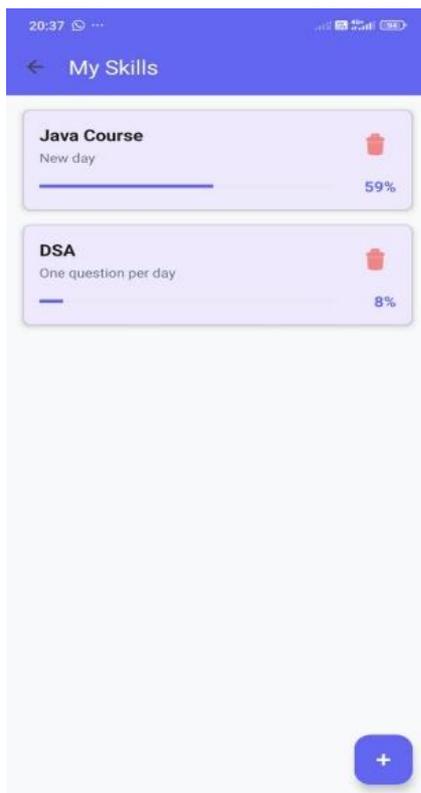


Figure 2: My Skills Page

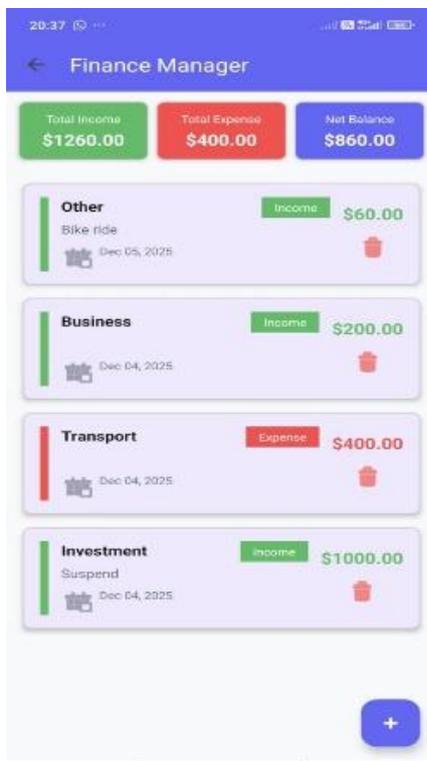


Figure 3: Finance Manager Page

VII. CONCLUSION

TaSkiFi demonstrates that a unified mobile application can effectively consolidate task management, skill

development, and personal finance tracking into a single, efficient platform. By adopting an offline-first architecture and dual storage approach using Room and Firestore, the system ensures uninterrupted access to essential user data even without network connectivity. The integration of owner backfilling and structured synchronization workflows further enhances data consistency across devices, making TaSkiFi reliable in real-world usage scenarios.

The implementation of a local reminder engine allows time-sensitive notifications to be delivered without dependence on the internet, addressing a common limitation in cloud-only productivity tools. Together, these design choices contribute to a cohesive, user-centered experience that reduces context switching and simplifies personal productivity management.

Future enhancements to TaSkiFi include support for Firebase Cloud Messaging to enable remote event-based alerts, advanced data insights such as charts and trends within each module, and the development of a web-based dashboard through REST APIs. These improvements will expand the platform's capabilities and provide users with a more comprehensive and accessible productivity ecosystem.

REFERENCES

- [1] Arya, A., & Sadhwani, M., "Cloud-Native Mobile Application Design Using Firebase and Serverless Backends," *International Journal of Mobile Computing and Application Research*, vol. 12, no. 3, pp. 45–52, 2023.
- [2] Banerjee, S., & Gupta, R., "Real-Time Synchronization Models for Distributed NoSQL Databases," *Journal of Cloud Information Systems*, vol. 18, no. 2, pp. 112–120, 2022.
- [3] Google Developers, "Firebase Authentication Documentation," 2024. Available: <https://firebase.google.com/docs/auth>
- [4] Google Cloud, "Cloud Firestore Documentation," 2024. Available: <https://firebase.google.com/docs/firestore>
- [5] Mishra, A., & Patel, K., "MVVM Architecture for Scalable Android Applications," *International Journal of Software Engineering and Systems*, vol. 9, no. 1, pp. 27–35, 2023.
- [6] Subramanian, L. N., & George, P., "User Engagement in Habit and Skill Tracking Applications," *ACM SIG Mobile Review*, vol. 25, no. 4, pp. 67–75, 2021.
- [7] Nakamura, T., "Design Considerations for Personal Finance Tracking on Cloud Platforms," *IEEE Access*, vol. 10, pp. 120865–120874, 2022.

- [8] Hasan, R., & Rahman, F., "Improving Mobile App Responsiveness Through Real-Time Data Streams," *Journal of Modern Mobile Computing*, vol. 14, no. 2, pp. 90–98, 2023.
- [9] Fowler, M., "Patterns for Mobile Application Synchronization," *ThoughtWorks*, 2023. Available: <https://martinfowler.com/articles/mobile-sync.html>
- [10] Krishnan, S., & Bhat, V., "Cross-Device Data Consistency in Cloud-First Mobile Applications," *International Journal of Emerging Computer Technologies*, vol. 17, no. 1, pp. 55–63, 2024.

Citation of this Article:

Chandrashekar G, Tharun Kumar N, Vivek M, Vikas, & Sangamesh G. Madagi. (2025). TaSkiFi – A Cloud-Synchronized Mobile System for Holistic Personal Productivity. *International Research Journal of Innovations in Engineering and Technology - IRJIET*, 9(12), 7-12. Article DOI <https://doi.org/10.47001/IRJIET/2025.912002>
