Al-Imam Muhammad Ibn Saud Islamic University

College of Computer and Information Sciences

Department of Computer Science

Smart Subscription Manager: Application for Managing User Subscriptions

By:
Raseel Aldowayan (443012322)
Sara Bedaiwi (444000149)
Alanoud Alyahya (444005874)
Jumanah Alorayyidh (444008567)

Supervisor: Dr. Sultan Alqahtani

Academic Year 2025 – Graduation Project 2 (GP2)

Table of Contents

Abbreviations

Abbreviation	Meaning
AI	Artificial Intelligence
ML	Machine Learning
NLP	Natural Language Processing
LSTM	Long Short-Term Memory
API	Application Programming Interface
FinTech	Financial Technology
UI	User Interface
UX	User Experience
DL	Deep Learning
ios	iPhone Operating System

1. System Design

1.1 Alternative Designs and Methods

Several alternative design approaches were evaluated during the system design phase, including system architecture, system decomposition, and algorithm selection. The evaluation criteria focused on scalability, maintainability, system complexity, and suitability for intelligent data processing.

A monolithic architecture was initially considered due to its simplicity; however, it was excluded because of limited scalability and maintenance difficulties. Consequently, a layered client–server architecture was adopted, as it provides better separation of concerns, supports system scalability, and facilitates integration with external services such as APIs and analytics modules.

For system decomposition, a pipeline (data-flow) model was evaluated but found unsuitable for systems requiring complex user interactions and persistent data management. Therefore, object-oriented decomposition (OOD) was selected, as it aligns with the system's object-based structure and enhances modularity and maintainability.

Regarding algorithm selection, traditional forecasting methods such as linear regression and ARIMA were considered; however, they were rejected due to their limited ability to model complex temporal patterns. Instead, the LSTM model was selected for its effectiveness in capturing long-term dependencies in time-series data [4]. Additionally, a rule-based approach for invoice data extraction was evaluated, but an NLP-based method was chosen due to its flexibility and robustness when handling varying email formats [6].

Table 1: Comparison of Alternative Designs and Selected Approaches

Aspect	Alternative Option	Limitations	Selected Approach	Justification
System Architecture	Monolithic Architecture	Limited scalability, high coupling, difficult maintenance	Layered Client–Server Architecture	Better separation of concerns, scalability, and easier integration with external services
System Decomposition	Pipeline (Data-Flow) Model	Not suitable for complex user interactions and persistent data	Object-Oriented Decomposition (OOD)	Improved modularity and maintainability
Forecasting Algorithm	Linear Regression / ARIMA	Limited ability to model complex temporal patterns	LSTM	Effective modeling of long-term dependencies in time-series data
Invoice Data Extraction	Rule-based (Regex) Extraction	Low flexibility and sensitive to email format changes	NLP-based Extraction	Robust and flexible handling of varying email formats

1.2 Graphical User Interface Design

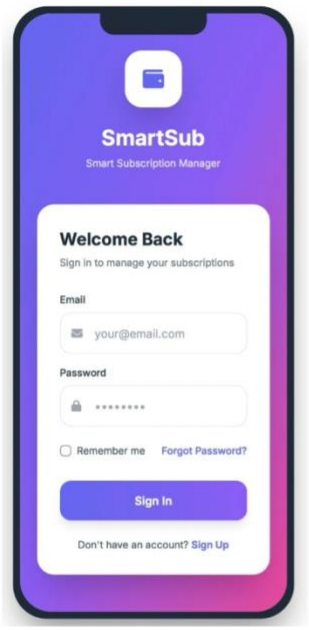

Figure 1: The login screen allows users to securely sign in to the SmartSub application.

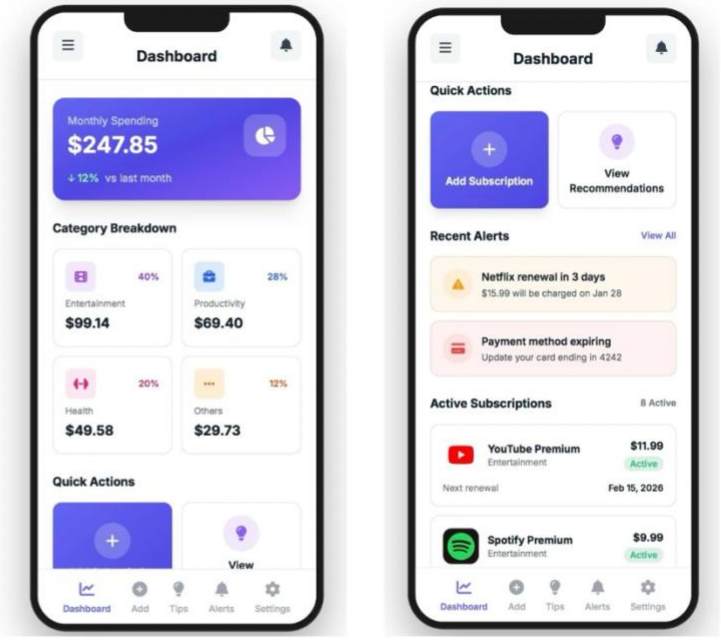

Figure 2: The dashboard screens present an overview of the user's subscription data, including monthly spending analytics, category breakdowns, quick actions, recent alerts, and a list of active subscriptions for efficient monitoring and management.

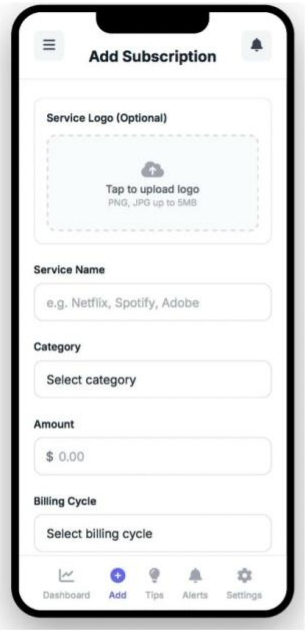

Figure 3: shows the Add Subscription screen, where users can enter subscription details such as the service name, category, amount, billing cycle, and optionally upload a service logo.

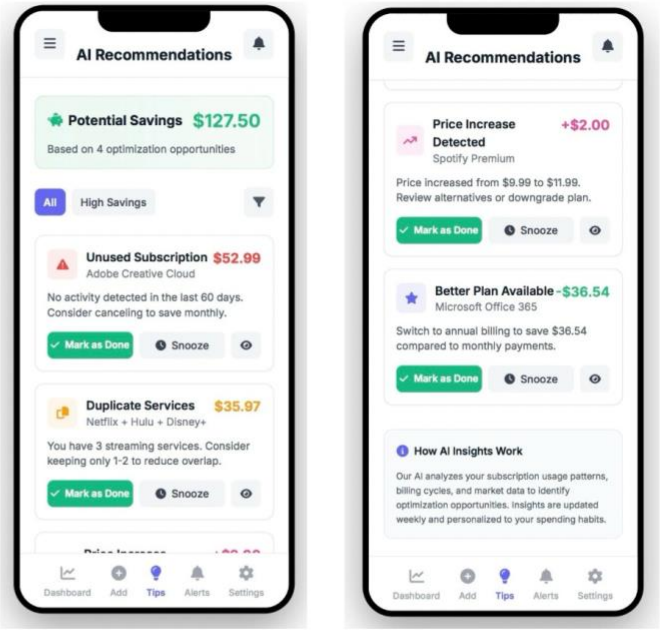

Figure 4: presents the AI Recommendations screens, highlighting price changes, unused or duplicate subscriptions, and potential savings, with quick actions to help users manage and optimize their subscriptions efficiently.

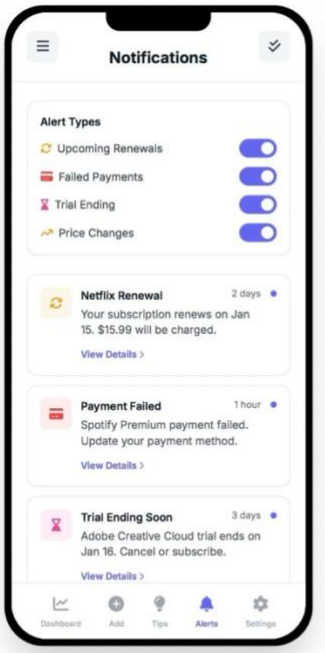

Figure 5: shows the Notifications screen, where users can manage alert preferences and view timely updates for renewals, failed payments, trials ending, and price changes.

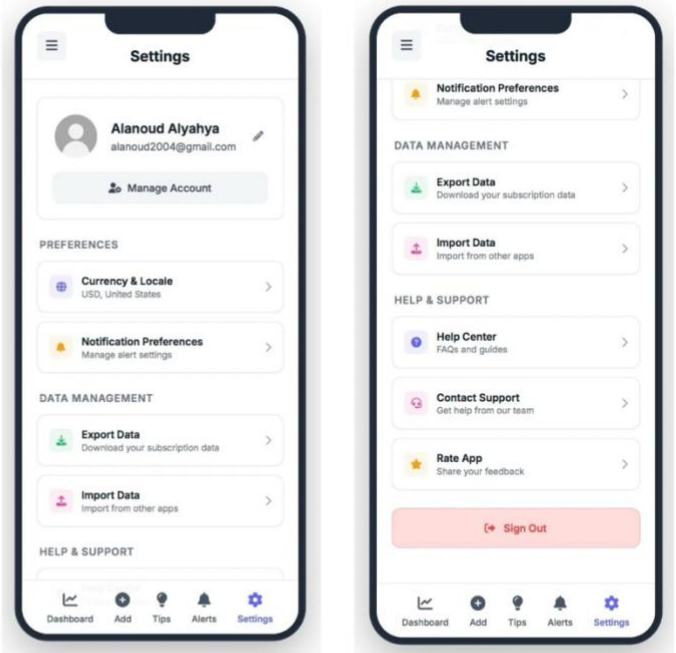

Figure 6: shows the Settings screens, where users can manage their account information, preferences, data import/export options, access help and support, and sign out of the application.

2. System Architecture

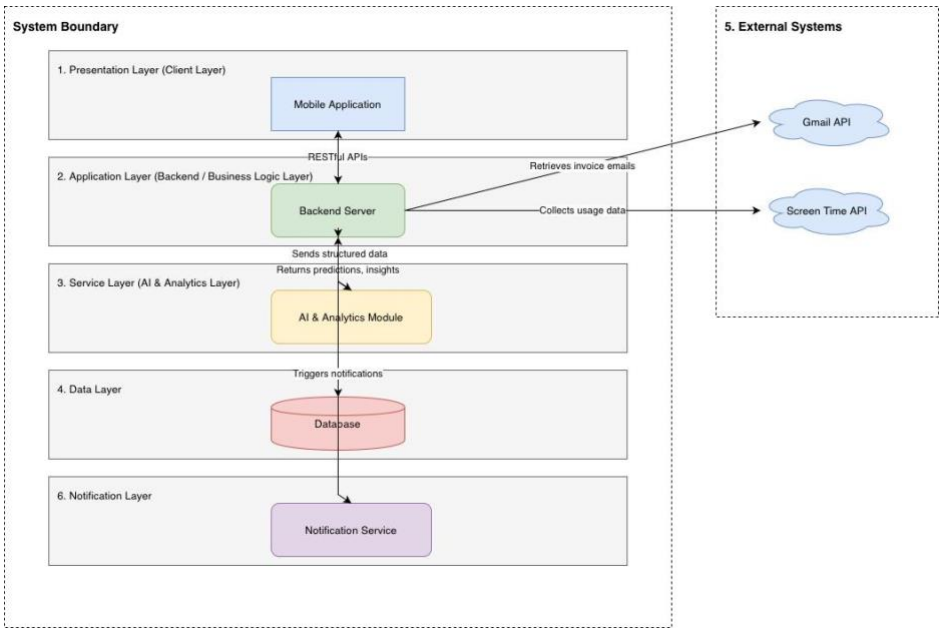

Figure 7: System architecture

The system architecture of the Smart Subscription Manager is designed using a layered architecture combined with a client–server model. This design separates system responsibilities into independent layers, improving scalability, maintainability, and system performance. The architecture consists of several interconnected subsystems that work together to collect subscription data, analyze user behavior, and provide intelligent insights and notifications.

The system is composed of the following main subsystems:

2.1 Presentation Layer (Client Layer)

The presentation layer represents the mobile application used by end users. It provides the graphical user interface through which users can register, log in, manage subscriptions, view dashboards, and receive recommendations and alerts. This layer sends user requests to the backend server through RESTful APIs and displays the processed results received from the system.

2.2 Application Layer (Backend / Business Logic Layer)

The application layer acts as the central controller of the system. It handles user requests, validates inputs, manages subscription data, and coordinates communication between internal components and external services. The backend server processes business logic such as subscription management, authentication, data synchronization, and API communication.

2.3 Service Layer (AI & Analytics Layer)

The service layer is responsible for intelligent data processing and analysis. It includes AI and analytics modules used for subscription classification, usage–cost analysis, and expense prediction. Natural Language Processing (NLP) techniques are applied to extract subscription information from invoice emails, while machine learning models analyze user behavior to generate personalized recommendations and forecasts.

2.4 Data Layer

The data layer manages persistent data storage. It stores user accounts, subscription records, invoice data, usage information, analytics results, and notification history. The database ensures data consistency, fast retrieval, and secure storage of all system information.

2.5 Notification Layer

The notification layer is responsible for delivering alerts and reminders to users. It generates renewal notifications, budget alerts, and recommendation messages based on system analysis and predefined rules.

2.6 External Systems Integration

The system integrates with external services to automate data collection. The Gmail API is used to retrieve invoice emails and detect subscription payments automatically. The Screen Time API is used to collect application usage data, allowing the system to evaluate subscription value by comparing usage with cost.

2.7Subsystem Communication

The system follows a layered client–server communication pattern. The mobile application communicates with the backend server through REST APIs. The backend server interacts with the database for data storage, communicates with external APIs for data retrieval, and sends processed data to the AI and analytics module. The generated results are returned to the mobile application and notification service for user interaction.

2.8 Design Justification

The layered architecture was selected because it separates user interface, business logic, analytics, and data management into independent components. This improves system maintainability, allows future expansion such as adding new APIs or AI models, and reduces system complexity.

3. Database Design

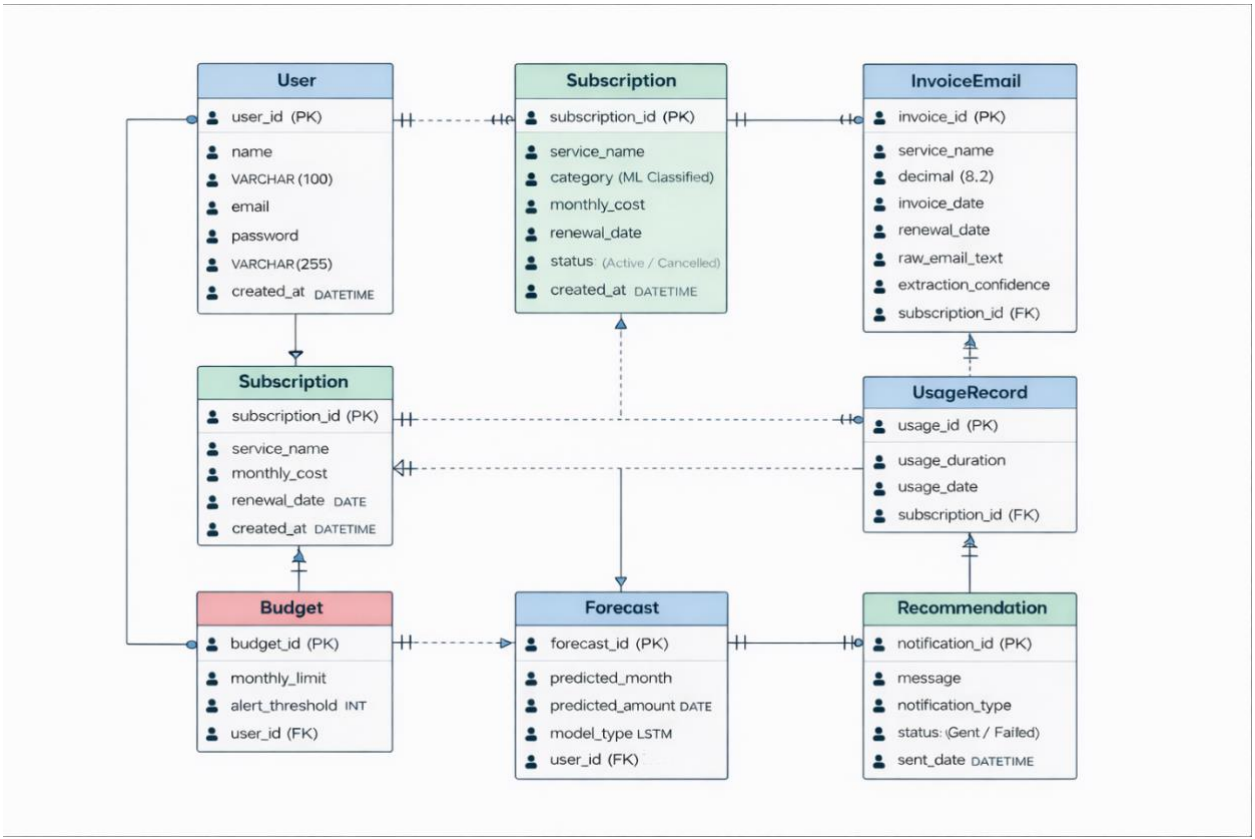

Figure 8: ER Diagram

Data Dictionary

1. User

Field	Type	Key	Description
user_id	INT	PK	Unique user identifier
name	VARCHAR(100)	—	User full name
email	VARCHAR(150)	Unique	User email
password	VARCHAR(255)	—	Hashed password
account_status	VARCHAR(20)	—	Active / Suspended
created_at	DATETIME	—	Account creation date

2. Subscription

Field	Type	Key	Description
subscription_id	INT	PK	Unique subscription ID
service_name	VARCHAR(100)	—	Service name
category	VARCHAR(50)	—	Category (ML classified)
monthly_cost	DECIMAL(8,2)	—	Monthly cost
renewal_date	DATE	—	Renewal date
status	VARCHAR(20)	—	Active / Cancelled
created_at	DATETIME	—	Added date
user_id	INT	FK	Reference to User

3. InvoiceEmail

Field	Type	Key	Description
invoice_id	INT	PK	Invoice identifier
service_name	VARCHAR(100)	—	Service name
amount	DECIMAL(8,2)	—	Invoice amount
invoice_date	DATE	—	Invoice date
renewal_date	DATE	—	Renewal date
raw_email_text	TEXT	—	Raw email content
extraction_confidence	DECIMAL(4,2)	—	NLP confidence score
subscription_id	INT	FK	Related subscription

4. UsageRecord

Field	Type	Key	Description
usage_id	INT	PK	Usage record ID
usage_duration	INT	—	Usage duration (minutes)
usage_date	DATE	—	Date
subscription_id	INT	FK	Related subscription

5. Budget

Field	Type	Key	Description
budget_id	INT	PK	Budget ID
monthly_limit	DECIMAL(8,2)	—	Monthly limit
alert_threshold	INT	—	Alert percentage
user_id	INT	FK	Reference to User

6. Forecast

Field	Type	Key	Description
forecast_id	INT	PK	Forecast ID
predicted_month	VARCHAR(20)	—	Forecasted month
predicted_amount	DECIMAL(8,2)	—	Predicted cost
model_type	VARCHAR(50)	—	LSTM
user_id	INT	FK	Reference to User

7. Recommendation

Field	Type	Key	Description
recommendation_id	INT	PK	Recommendation ID
recommendation_type	VARCHAR(50)	—	Cancel / Keep / Downgrade
reason	TEXT	—	Recommendation reason
confidence_score	DECIMAL(4,2)	—	AI confidence
subscription_id	INT	FK	Related subscription

8. Notification

Field	Type	Key	Description
notification_id	INT	PK	Notification ID
message	TEXT	—	Notification text
notification_type	VARCHAR(50)	—	Renewal / Budget
status	VARCHAR(20)	—	Sent / Failed
sent_date	DATETIME	—	Sent time
user_id	INT	FK	Reference to User

4. References

[1] C+R Research, “Subscription Service Statistics and Costs,” C+R Research Reports, 2022. [Online]. Available: https://www.crresearch.com/blog/subscription-service-statistics-and-costs/

[2] L. Cao, “AI in Finance: Challenges, Techniques and Opportunities,” arXiv preprint arXiv:2107.09051, 2021. [Online]. Available: https://arxiv.org/abs/2107.09051

[3] S. Minaee, N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu, and J. Gao, “Deep learning–based text classification: A comprehensive review,” ACM Computing Surveys (CSUR), vol. 54, no. 3, pp. 1–40, 2022. [Online]. Available: https://dl.acm.org/doi/10.1145/3439726

[4] X. Liu, L. Wang, and Y. Zhang, “Deep Time Series Forecasting Models: A Comprehensive Survey,” Mathematics, vol. 12, no. 10, p. 1504, 2024. [Online]. Available: https://www.mdpi.com/2227-7390/12/10/1504

[5] B. Khattak et al., “Customer churn prediction using composite deep learning technique,” BMC Medical Informatics and Decision Making, vol. 23, no. 1, 2023. [Online]. Available: https://pmc.ncbi.nlm.nih.gov/articles/PMC10570272/

[6] N. Malik et al., “Natural language processing for analysing online customer reviews,” J. Clin. Transl. Endocrinol., vol. 36, 2024. [Online]. Available: https://pmc.ncbi.nlm.nih.gov/articles/PMC11323031/

[7] A. S. Chougule et al., “Implementation of Subscription Management SaaS-Based System,” IJRASET, 2022. [Online]. Available: https://www.ijraset.com/research-paper/implementation-of-subscription-management-saas-based-system

[8] M. Imani, M. Joudaki, A. Beikmohammadi, and H. R. Arabnia, “Customer Churn Prediction: A Review of Recent Advances, Trends, and Challenges in Conventional Machine Learning and Deep Learning,” preprint, Mar. 2025. [Online]. Available: https://www.preprints.org/manuscript/202503.1969/v1

[9] A. Amari et al., “An Efficient Deep Learning-Based Approach to Automating Invoice Document Validation,” arXiv preprint, Mar. 2025. [Online]. Available: https://arxiv.org/abs/2503.12267

[10] Rocket Money – Find & Cancel Subscriptions. [Online]. Available: https://www.rocketmoney.com/

[11] StayFree – Screen Time Tracker & Overuse Reminder. [Online]. Available: https://stayfreeapps.com/