

# COLLEGE OF COMPUTING AND INFORMATICS

## Department of Software Engineering



Year III- semester 1

Web programming 2

Name	ID
1. Tolosa Tesfaye .....	1010/15
2. Tesfaye Eshatu.....	0996/15
3. Tewodiros Fikadu ....	1000/15
4. Worku Teshome.....	1036/15

Submission to: Fufa W

Submission date: 29 jun 2025

HARAMAYA | OROMIA | ETHIOPIA

<i><b>Table content</b></i>	<b>page</b>
➤ cover page.....	1
➤ Introduction and table content .....	2
➤ Project Title and Group Number.....	3
➤ Team Profile.....	4
➤ Project Description.....	4
➤ Project objective.....	5
➤ Services Provided by the Proposed System.....	6
➤ Plan of Work.....	7
➤ Resource Requirements and Cost Estimates.....	9
➤ functional Metrics.....	9
➤ conclusion.....	10

## Introduction

The Subscription Management System project aims to streamline the management of subscription-based services for businesses, addressing common challenges like scalability, payment failures, data security, and customer management. With the growing trend of subscription models in various industries (e.g., SaaS, streaming, retail), the system will provide a scalable platform to efficiently track and manage subscriptions, automate billing processes, and ensure secure payment handling. The project will also focus on improving customer engagement through accurate data management and user-friendly interfaces, all while maintaining a high level of system performance and security. This solution is designed to help businesses reduce revenue leakage, minimize churn, and enhance customer satisfaction.

# Project title: Subscription Management System proposal

Our group are 4 members:

👉 Tolesa Tesfaye



➡ *Role: Project Manager*

• *Strengths: Tolesa exhibits strong leadership and organizational skills, coupled with effective communication abilities. With experience in project planning and management, Tolesa is adept at coordinating team efforts, setting achievable goals, and ensuring that project milestones are met on time. As the elected Team Leader, Tolesa oversees the project's progression and facilitates collaboration among team members*

👉 Tesfaye Eshatu



➡ *Role: Backend Developer*

• *Strengths: Tesfaye is proficient in server-side programming, database management, and API development. His expertise lies in ensuring application scalability and performance. Tesfaye is adept at building robust backend systems that form the core foundation of the subscription management platform, focusing on reliability and efficiency.*

👉 Tewodiros Fikadu



➡ *Role: Frontend Developer*

• *Strengths: Tedy specializes in creating intuitive user interfaces with a keen eye for responsive design and enhancing user experience. Skilled in HTML, CSS, JavaScript, and modern frontend frameworks, Tedy is responsible for developing interfaces that are both aesthetically pleasing and user-friendly, ensuring a seamless interaction for end-users*

👉 Worku Teshome



➡ *Role: Quality Assurance (QA) Engineer*

• *Strengths: Worku is detail-oriented with a strong focus on software testing and quality assurance. Proficient in developing and executing test plans and cases, Worku excels at identifying bugs and ensuring that the product meets the highest quality standards. His meticulous approach contributes to the reliability and stability of the system.*

## ☑ Project Description

The subscription management system addresses the growing number of businesses that offer subscription-based products and services. These may include software as a service (SaaS), streaming services, retail subscriptions, and more. Managing subscriptions manually can be cumbersome, error-prone, and inefficient, leading to revenue leakage, customer dissatisfaction, and operational complexities

✓ **Some potential issues and challenges has below.**

### ☑ Scalability Concerns:

As the number of users and their subscriptions grow, the system may struggle to handle increased data volume and user requests, leading to performance degradation

### ☑ Data security:

Financial data security can be a real issue when it comes to software subscription management. This can be especially problematic if you pay via credit card standing instructions.

### ☑ Handling Payment Failures and inconsistent billing :

Failed payments due to expired or invalid payment methods can lead to revenue loss and require effective dunning processes to manage And also Inaccurate billing cycles and missed payments can result in financial losses and affect customer trust.

### ☑ Customer Management:

Poor customer management leads to high churn rates as businesses struggle to maintain accurate and up-to-date records.

## ☑ Project objectives

### 1. Develop a Scalable System:




- Design and implement a platform that can efficiently handle growing user bases and increasing subscription data volume, ensuring system performance remains optimal even with high traffic.

### 2. Ensure Robust Data Security

- Implement strong encryption and security protocols to safeguard sensitive financial and personal information, particularly when processing subscription payments via credit cards and other financial methods.
3. **Optimize Payment and Billing Management:**
- Build reliable mechanisms for managing payment failures, invalid payment methods, and billing discrepancies to ensure accurate and timely processing of subscriptions, with appropriate dunning and recovery processes.
4. **Improve Customer Management:**
- Develop features for maintaining accurate and up-to-date customer records, enabling businesses to engage customers effectively and reduce churn rates by providing personalized experiences and communication.
5. **Enhance User Experience:**
- Create an intuitive and user-friendly interface that allows users to easily manage their subscriptions, view payment history, update details, and receive notifications about important billing events or changes.
6. **Focus on Quality and Stability:**
- Ensure the system undergoes thorough testing, including functional, integration, and stress testing, to deliver a reliable, bug-free platform that meets both technical and user expectations.

## **Services Provided by the Proposed System**

The Subscription Management System will help businesses efficiently manage subscription-based services, addressing challenges such as scalability, payment issues, data security, and customer engagement.

-  **Subscription Tracking:** Track and manage customer subscriptions, renewal dates, and payment history.
-  **Billing & Payment Management:** Automate billing cycles, handle failed payments, and ensure secure payment processing.
-  **Customer Management:** Manage customer data and reduce churn through proactive engagement.

- ✦ Security & Data Protection: Use encryption and secure authentication to protect sensitive data.
- ✦ Scalability: Handle increasing users and subscriptions without performance issues.

### **\* New Programs or Existing Tools**

- New Programs: Custom solutions for subscription tracking, payment management, and customer handling.
- Existing Tools: Integration of third-party services like payment gateways (Stripe, PayPal) and analytics tools.

#### System's Functionalities

- User Registration & Authentication: Secure login and account setup.
- Subscription Plan Management: Manage various subscription plans.
- Payment Gateway Integration: Process payments securely.
- Automated Billing: Generate and send invoices.
- Dunning Process: Handle failed payments.
- Customer Communication: Send notifications for renewals and failures.

### **\* Potential or Actual Customers**

- ❖ SaaS Providers
- ❖ Media Streaming Services
- ❖ E-commerce & Retail Subscription Boxes
- ❖ Educational Platforms
- ❖ Fitness & Health Clubs

## **☑ Plan of Work for the Subscription Management System**

### Short-Term Deliverables (Next Few Weeks)

1. Week 1: Requirement Analysis and Initial Setup
  - Tolesa (Project Manager): Coordinate the gathering of detailed system requirements from potential customers, and define the project timeline.

- Team: Set up the development environment, version control systems (Git), and project management tools
2. Week 2: Database and Backend Setup
- Tesfaye (Backend Developer): Design the database schema for subscription tracking, user management, billing, and payments. Implement data storage mechanisms for customer records and subscription data.
  - Worku (QA): Create initial test cases to verify data integrity and security protocols.
3. Week 3: Customer Registration and Authentication
- Tewodiros (Frontend Developer): Develop user registration and login interface with secure authentication methods (e.g., password hashing )
  - Tesfaye (Backend Developer): Implement the API to handle user registration and authentication, linking it to the database for storing credentials.
  - Worku (QA): Test the registration and login process, checking for security vulnerabilities.
4. Week 4: Subscription Plan Management
- Tesfaye (Backend Developer): Develop API endpoints for managing subscription plans (add, modify, delete) and associating them with user accounts.
  - Tewodiros (Frontend Developer): Build the interface for displaying and allowing users to choose from available subscription plans.
  - Worku (QA): Test the functionality for subscription plan management.

### **Team Subgroups and Task Assignment**

1. Backend Development (Tesfaye + Worku)
  - Tasks:
    - Database schema design and implementation
    - API development for user management, subscription plans, payment integration, and billing
    - Security implementation (data encryption, secure transactions)
    - Payment failure handling and dunning process
2. Frontend Development (Tewodiros + Worku)

- Tasks:
  - User registration and login interface
  - Subscription plan display and selection interface
  - Payment interface and transaction confirmation UI
  - Handling of payment failure notifications and error messages

### 3. Quality Assurance (Worku + Tolesa)

- Tasks:
  - Writing test cases for all critical system functionalities (authentication, subscription management, payment processing)
  - Manual and automated testing of features as they are developed
  - Coordination with developers to report issues and ensure timely fixes

The key focus for the upcoming weeks will be on building core functionalities like user registration, subscription management, and payment processing.

#### Team Strengths and Suitability for the Project

- Project Manager: Responsible for overseeing the project and ensuring timelines are met.
  - Strength: Experienced in managing academic projects, ensuring effective coordination among team members.
- System Architect: Designs the technical infrastructure and system components.
  - Strength: Knowledgeable about system design for academic project environments and cloud-based platforms.
- Backend Developers: Develop core system features.
  - Strength: Understanding of backend technologies like Node.js, Python, and SQL databases in a class project context.
- Frontend Developers: Design and develop the user interface.
  - Strength: Skilled in creating user-friendly UIs with React.js and Vue.js, appropriate for academic projects.

### Resource Requirements and Cost Estimates



## ☑Resources Needed:

### 💰 Software Tools:

- Project management tools
- Development tools (e.g., GitHub, AWS)
- Testing and deployment tools

### 💰 Physical and Material Resources

Items required to support development:

- **Laptops or Desktops:** For coding, testing, and documentation
- **Internet Connection:** Reliable internet for team collaboration and access to online resources
- **Storage Devices:** For backups and sharing files (e.g., USB drives or cloud storage)

### 💰 Time Resources

Estimated time allocation for each phase:

- ✓ **Planning and Design:** 1 week
- ✓ **Backend Development:** 1 weeks
- ✓ **Frontend Development:** 1 weeks
- ✓ **Integration and Testing:** 3 day.
- ✓ **Final Report and Presentation:** 4 day.

### 💰 Other Resources

✓**Documentation:**

- Microsoft Word, Google Docs, or LaTeX for writing reports and proposals
- Canva or Figma for visual designs and wireframes

## 🌟 Success Metrics for Evaluating the Project Outcome

### → *Functional Metrics*

User Authentication:

- Users can successfully register and log in with secure authentication (e.g., hashed passwords).
- Login errors (e.g., invalid credentials) are correctly handled.

Subscription Plan Management:

- Users can view available subscription plans and subscribe without errors.

- Admins (or developers, for class purposes) can add, modify, or delete subscription plans via backend functionality.  
→ *Quality Assurance Metrics*
- Error Handling:
  - The system gracefully handles common errors, such as invalid inputs, failed payments, or database connectivity issues.
- Testing Coverage:
  - At least 80% of key functionality (e.g., registration, subscriptions, and payments) is tested and verified by QA

## Conclusion

The Subscription Management System (SMS) proposal demonstrates a robust and innovative solution tailored to address the challenges faced by subscription-based businesses, including scalability, data security, payment management, and customer retention. By leveraging the combined expertise of the team, the project is structured to deliver a high-quality platform that meets modern subscription management needs.

Each team member plays a pivotal role: Tolesa Tesfaye ensures effective project coordination as the Project Manager; Tesfaye Eshatu focuses on building a scalable and efficient backend; Tewodios Fikadu creates a seamless and user-friendly frontend; and Worku Teshome ensures quality through rigorous testing. This well-defined division of roles and responsibilities ensures a balanced approach to development, quality, and delivery.

The SMS aims to deliver core functionalities, such as subscription tracking, automated billing, secure payment processing, and proactive customer management. The proposed system emphasizes robust data security, reliable payment mechanisms, and intuitive user interfaces, ensuring a positive user experience while maintaining operational excellence.

The detailed project timeline outlines a systematic approach to development, starting with requirement analysis, backend and frontend development, and rigorous quality assurance testing. The inclusion of scalable architecture, integration with third-party tools like Stripe and PayPal, and features such as dunning processes and automated billing demonstrates the project's focus on practical, real-world application.

In summary, the Subscription Management System is a thoughtfully designed, innovative project with the potential to streamline subscription management processes, reduce customer churn, and improve business efficiency.