Software Requirements Specification
For
Team Merge Conflict

Application: Recur

September 22, 2025

Version 2

Prepared by:
Curtis Lemke

Table of Contents

Revision History

Version	Date	Name	Description
1	09/22/25	Curtis Lemke	Initial Document
2	12/09/25	Curtis Lemke	Final revisions

1. Introduction

1.1 Overview

The **Recur application** is a web-based subscription and recurring expense management platform available to users through modern web browsers on both desktop and mobile devices. The application provides users with tools to track, analyze, and manage recurring financial commitments such as subscriptions, memberships, and regularly billed services.

Recur integrates with financial data sources to automatically identify recurring transactions and allows users to manually manage subscriptions through direct entry or CSV import. The application presents upcoming renewals, historical spending analytics, and categorized insights to help users better understand and control their recurring expenses.

This document provides the requirements for the Recur software application. Project goals, scope, and definitions are provided in the introduction. Design constraints and application environment are described in the following section. Non-functional requirements are outlined for later verification. Functional requirements describe system features and expected user interaction.

Project constraints such as budget and schedule are addressed in a separate Software Project Management Plan. A separate Test Plan document specifies test strategies, procedures, and acceptance criteria.

---

2. Goals and Objectives

The primary objective of the Recur project is to provide users with a centralized, intuitive way to track and manage recurring financial obligations.

The Recur application is expected to:

- Provide a clear interface for viewing recurring subscriptions and expenses
- Automatically detect recurring transactions from financial data
- Allow manual subscription management through user input or CSV import
- Present upcoming renewals and historical analytics
- Function in a simple, intuitive, and visually consistent manner

---

3. Scope

The Recur application enables users to:

- Create and authenticate an account
- Connect financial accounts through third-party integrations
- View transaction history and detect recurring expenses
- Track subscriptions, renewal dates, and costs
- View analytics such as monthly spending and category breakdowns
- Manually manage subscriptions without linking a bank account

The application is delivered as a web application and does not require installation. Financial data access is optional, and users may choose to use manual data entry features exclusively.

---

4. Definitions

- **Recur Application** – The software system described in this document.

- **Project** – Activities leading to the design, development, and deployment of Recur.
- **Client** – The organization or individual for whom Recur is developed.
- **User** – An individual who interacts with the Recur application.
- **Use Case** – A goal-oriented interaction between a user and the system.
- **Scenario** – A single path through a use case.
- **Actor** – A user or external system interacting with Recur.
- **Developer** – The individual or organization developing the application.
- **Stakeholder** – Any individual or group with an interest in the project and its outcomes.

---

5. General Design Constraints

5.1 Application Environment

The Recur application is a web-based system built using a modern TypeScript framework. The frontend interfaces with backend services responsible for authentication, data storage, and financial data aggregation.

The system integrates with:

- A backend database for persistent storage
- Authentication services for user identity management
- External financial APIs for transaction ingestion

This architecture allows for scalability, maintainability, and separation of concerns.

---

6. User Characteristics

Recur Users are individuals seeking to better manage personal finances and recurring expenses. Users are expected to have basic familiarity with web applications and financial concepts such as subscriptions and monthly billing. No advanced technical expertise is assumed.

---

7. Mandated Constraints

- The application must run in modern web browsers
- The system must support both desktop and mobile layouts
- User data must be stored securely using industry-standard practices

---

8. Nonfunctional Requirements

8.1 Operational Requirements

Usability:
At least 95% of users should be able to use Recur without reading documentation or requiring training.

8.2 Performance Requirements

- Page loads and core interactions should complete within acceptable response times under normal network conditions.

8.3 Maintainability

- Changes to external financial APIs should be accommodated without requiring changes to the frontend user interface.
- Modular architecture should allow isolated updates to individual system components.

8.4 Security Requirements

- User authentication must be handled securely.
- Sensitive credentials must never be stored in plaintext.
- Financial data must be transmitted and stored using secure, encrypted channels.

---

9. Documentation and Training

The Recur application is delivered as a web service and does not require user training. Online help text and system documentation will be provided to stakeholders and developers.

---

10. External Interface Requirements

10.1 User Interface

The user interface will be visually consistent, modern, and minimalistic. It will emphasize clarity, ease of navigation, and financial insight at a glance.

The interface must be intuitive, requiring no formal training. The system should guide users naturally through common tasks such as viewing subscriptions or analytics.

10.2 Software Interface

The frontend communicates with backend services via secure APIs. External financial data providers serve as additional interfaces for transaction ingestion and recurring payment detection.

---

11. Functional Requirements

11.1 Required Features

Use Case 1: User Login and Dashboard Access

Actors: Registered user
Value: High
Cost: High

Basic Path

1. User navigates to the Recur application.
2. System prompts the user to log in or sign up.
3. User enters valid credentials.
4. System authenticates the user.
5. System displays the dashboard with subscriptions, upcoming renewals, and summary analytics.
6. User logs out.
7. System ends session.

Alternate Path

- Invalid credentials result in an error message.
- User may retry login or exit the application.

Use Case 2: View Subscriptions and Upcoming Renewals

Actors: Registered user
Value: High
Cost: Medium

Basic Path

1. User accesses the dashboard.
2. System displays a list of detected and manually added subscriptions.
3. System shows renewal dates and recurring costs.

4. User exits the view.

11.2 Optional Features

Use Case 3: View Analytics and Historical Spending

Actors: Registered user
Value: Medium
Cost: High

Basic Path

1. User navigates to the analytics section.
2. System displays charts showing spending by category and by month.
3. User returns to dashboard or logs out.