

# Software Design Document

## Subscription Management and Tracking App

Prepared by: Abdulwahab Alnemer, Yosef Alhedr

09/09/2023

## Table of Contents

1. Introduction.....	3
2. System Architecture .....	3
2.1 Classes and Methods.....	4
3. GUI Sketch.....	6
3.1 Sign-in / Sign-up page .....	6
3.3 Notification Center.....	6
4. ER Diagram .....	7
4.1 Tables .....	7
4.2 Keys .....	7

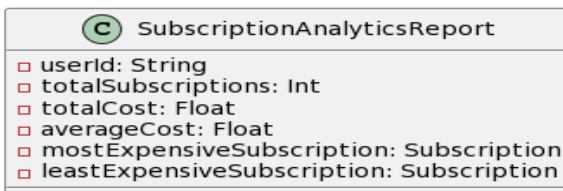
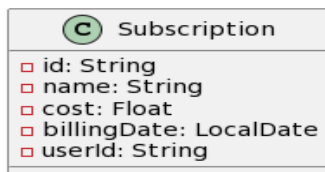
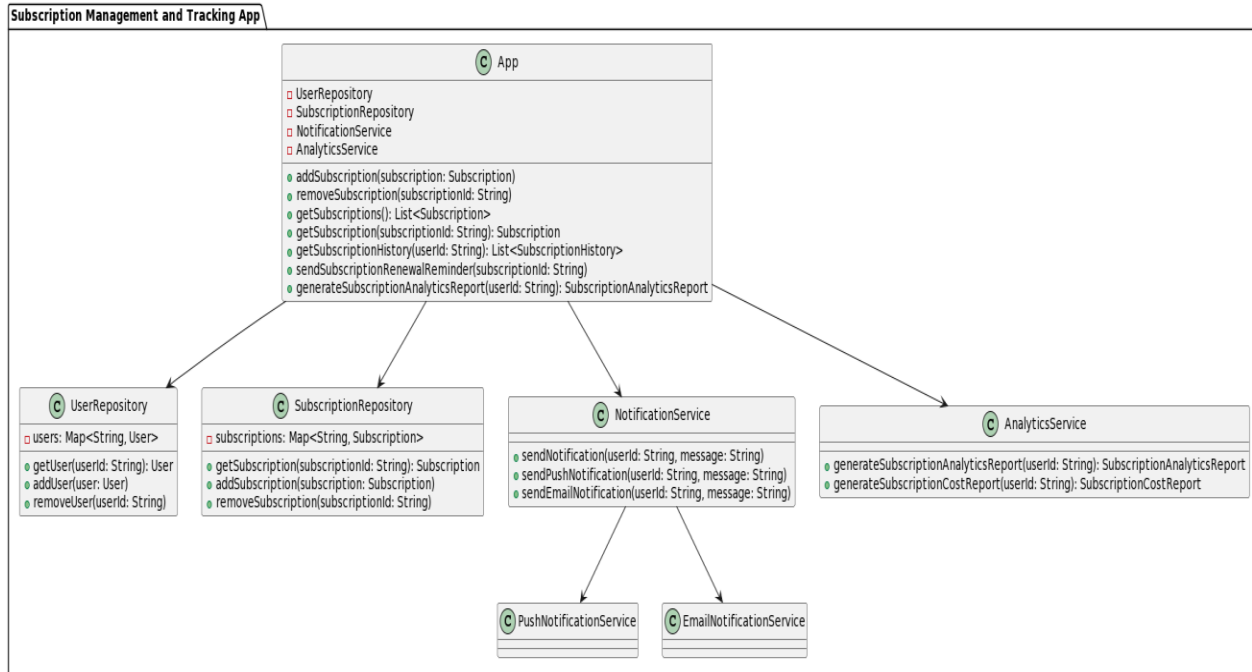
## Revision History

Name	Date	Reason For Changes	Version

## 1. Introduction

The Subscription Management and Tracking App is designed to simplify and streamline the management of user subscriptions. This document shows the design for the app, including diagrams, classes, and tables.

## 2. System Architecture



### 2.1 Classes and Methods

**App class:** This is the main entry point for the application. It interacts with the UserRepository, SubscriptionRepository, NotificationService, and AnalyticsService classes to perform tasks such as adding and removing subscriptions, sending renewal reminders, and generating analytics reports.

- **Methods:**
  - addSubscription(subscription: Subscription): This method adds a new subscription to the system.
  - removeSubscription(subscriptionId: String): This method removes an existing subscription from the system.
  - getSubscriptions(): List<Subscription>: This method returns a list of all subscriptions for the current user.
  - getSubscription(subscriptionId: String): Subscription: This method returns the subscription with the specified ID.
  - getSubscriptionHistory(userId: String): List<SubscriptionHistory>: This method returns a list of all subscription history records for the current user.
  - sendSubscriptionRenewalReminder(subscriptionId: String): This method sends a renewal reminder notification to the user for the subscription with the specified ID.
  - generateSubscriptionAnalyticsReport(userId: String): SubscriptionAnalyticsReport: This method generates a subscription analytics report for the current user.

**UserRepository class:** This class stores information about users, such as their name, email address, and subscription history.

- **Methods:**
  - getUser(userId: String): User: This method returns the user with the specified ID.
  - addUser(user: User): This method adds a new user to the system.
  - removeUser(userId: String): This method removes an existing user from the system.

**SubscriptionRepository class:** This class stores information about subscriptions, such as their name, cost, and billing date.

- **Methods:**
  - getSubscription(subscriptionId: String): Subscription: This method returns the subscription with the specified ID.
  - addSubscription(subscription: Subscription): This method adds a new subscription to the system.
  - removeSubscription(subscriptionId: String): This method removes an existing subscription from the system.

**NotificationService class:** This class is responsible for sending notifications to users, such as renewal reminders.

- **Methods:**
  - sendNotification(userId: String, message: String): This method sends a notification to the user with the specified ID.
  - sendPushNotification(userId: String, message: String): This method sends a push notification to the user with the specified ID.

## Software Design Document for Subscription Management and Tracking App

- `SendEmailNotification(userId: String, message: String)`: This method sends an email notification to the user with the specified ID.

**AnalyticsService class:** This class generates reports on user subscriptions, such as the total number of subscriptions, the total cost, and the average cost.

- **Methods:**
  - `generateSubscriptionAnalyticsReport(userId: String)`: `SubscriptionAnalyticsReport`: This method generates a subscription analytics report for the current user.
  - `generateSubscriptionCostReport(userId: String)`: `SubscriptionCostReport`: This method generates a subscription cost report for the current user.

**Subscription class:** This class represents a subscription. It stores information such as the name, cost, and billing date of the subscription.

- **Attributes:**
  - `id`: The ID of the subscription.
  - `name`: The name of the subscription.
  - `cost`: The cost of the subscription.
  - `billingDate`: The billing date of the subscription.
  - `userId`: The ID of the user who owns the subscription.

**SubscriptionHistory class:** This class represents a subscription history record. It stores information such as the ID of the subscription, the action that was performed, and the date of the action.

- **Attributes:**
  - `subscriptionId`: The ID of the subscription.
  - `action`: The action that was performed.
  - `date`: The date of the action.

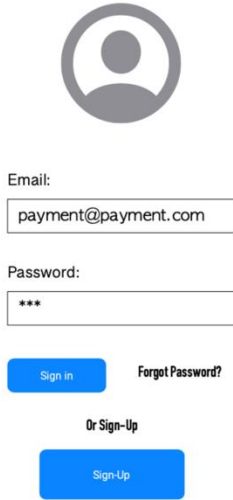
**SubscriptionAnalyticsReport class:** This class represents a subscription analytics report. It stores information such as the total number of subscriptions, the total cost, and the average cost for the user.

- **Attributes:**
  - `userId`: The ID of the user.
  - `totalSubscriptions`: The total number of subscriptions.
  - `totalCost`: The total cost of all subscriptions.
  - `averageCost`: The average cost per subscription.
  - `mostExpensiveSubscription`: The most expensive subscription.
  - `leastExpensiveSubscription`: The least expensive subscription.

# Software Design Document for Subscription Management and Tracking App

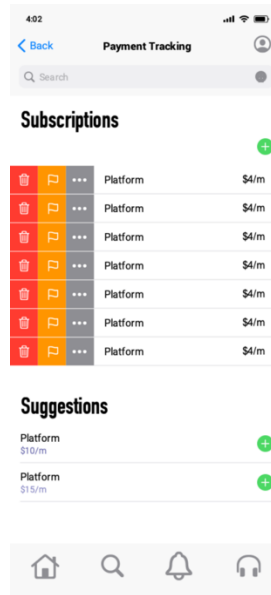
## 3. GUI Sketch

### 3.1 Sign-in / Sign-up page

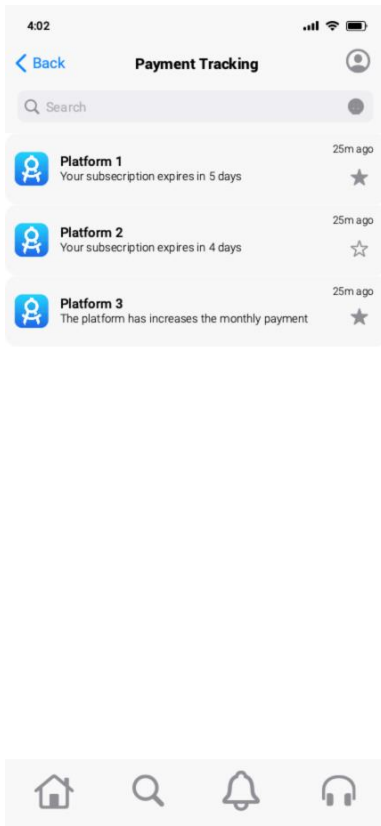


A sketch of a sign-in/sign-up page. At the top is a circular profile icon placeholder. Below it are two input fields: 'Email:' with the text 'payment@payment.com' and 'Password:' with three asterisks. A blue 'Sign in' button is next to the password field, and a 'Forgot Password?' link is to its right. Below these is the text 'Or Sign-Up' and a blue 'Sign-Up' button.

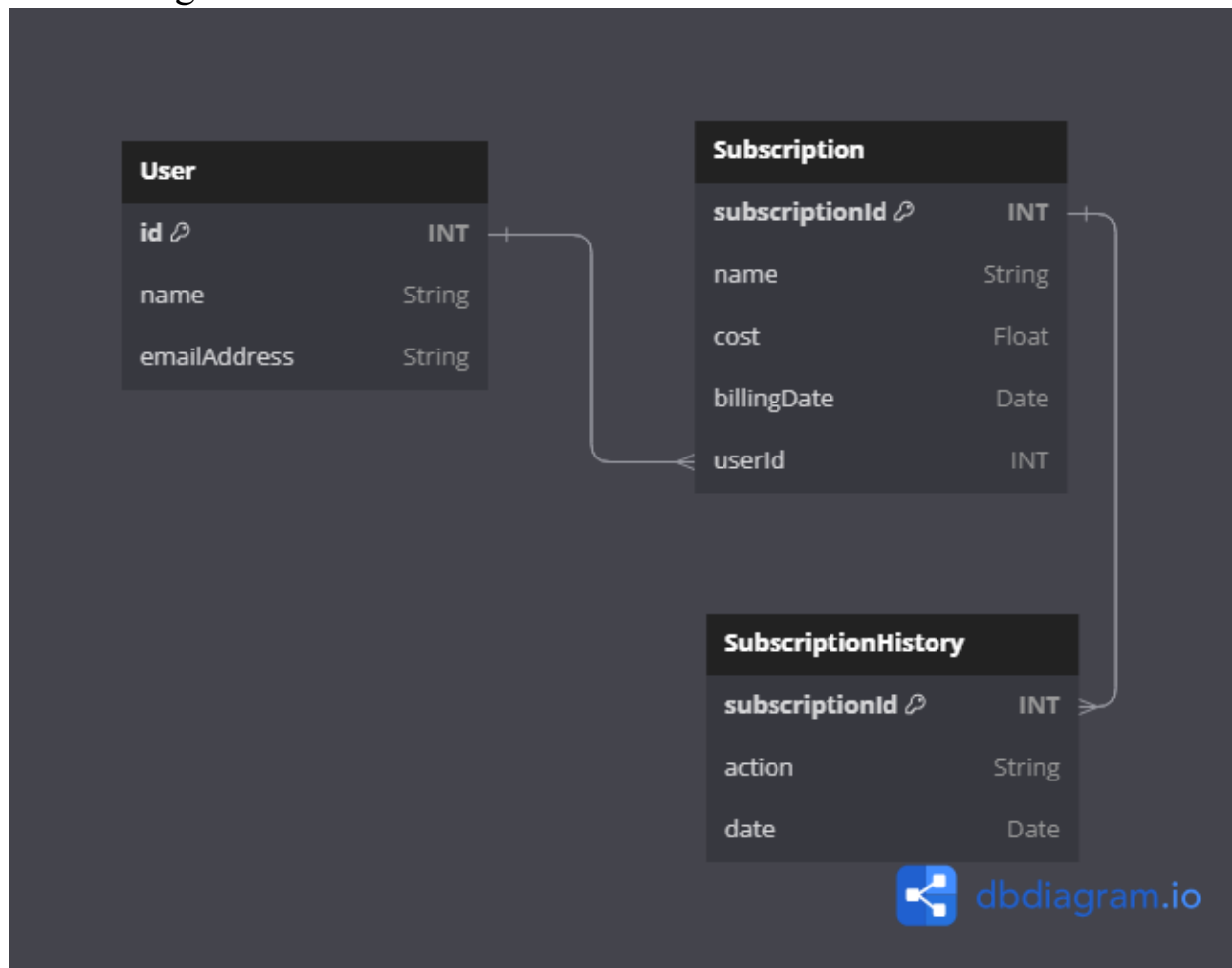
### 3.2 Home Page



### 3.3 Notification Center



## 4. ER Diagram



### 4.1 Tables

**User Table:** This table stores information about users, such as their ID, name, and email address.

**Subscription Table:** This table stores information about subscriptions, such as their ID, name, cost, billing date, and user ID.

**SubscriptionHistory Table:** This table stores information about subscription history records, such as the ID of the subscription, the action that was performed, and the date of the action.

### 4.2 Keys

**Primary Keys:**

**User Table:** The primary key is the userId column.

## **Software Design Document for Subscription Management and Tracking App**

Subscription Table: The primary key is the subscriptionId column.

SubscriptionHistory Table: The primary key is the subscriptionId column.

Foreign Keys:

Subscription Table: The userId column is a foreign key that references the userId column in the User Table.

SubscriptionHistory Table: The subscriptionId column is a foreign key that references the subscriptionId column in the Subscription Table.