Software Architecture Document

By: Nidhi Patel
Elishbah Younus
Zan Peracha
Pamela Shahu
Ahmed Alfaris

1)Introduction

Purpose

This is the architectural document for the SplitSmart Project. Users of SplitSmart will be able to establish and keep track of spending as well as pay off any outstanding balance for splitting costs with friends, roommates, family, coworkers, or other individuals or groups of people. This document provides an overview of the architecture's goals, constraints, components, and interactions.

Scope

This document includes the design of the SplitSmart system. It focuses on the different aspects of the system like creating groups and new expenses by the users, generating reports, and adding new members to the groups. It also includes integrating the functionality of tracking balances and payments made by other users.

2) Architectural Representation

This document presents the architecture as a series of views; use case view, logical view, process view and deployment view implementation view. These are views on an underlying Unified Modeling Language (UML) model developed using Rational Rose.

3) Architectural Goals and Constraint

Some architectural goals and constraints are as follows:

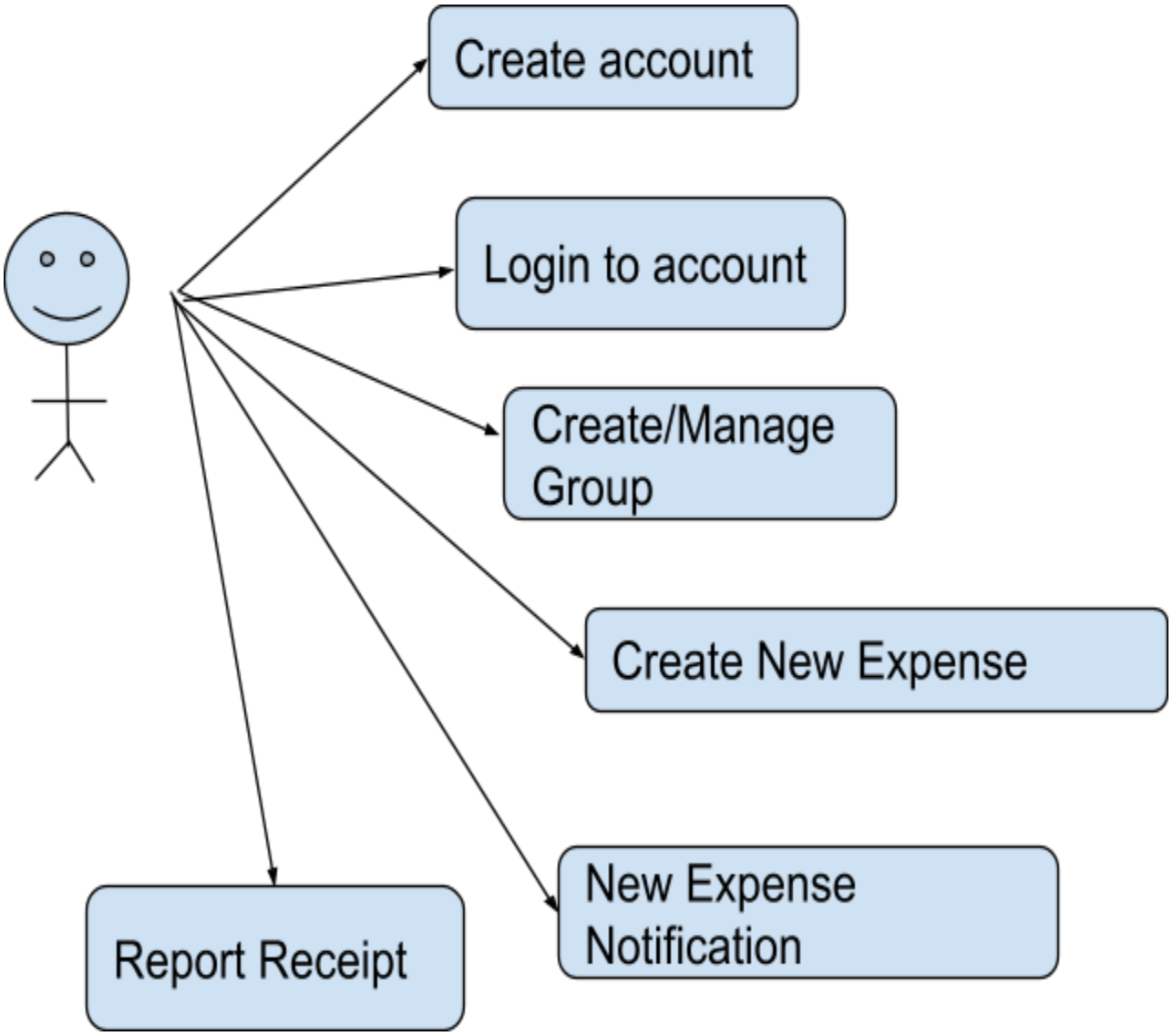
- The system architecture should be capable enough to handle the growing volume of users and it should not affect the system's performance.
- The architecture should be flexible to add more functionality to the system later on.
- The architecture must be able to integrate different plugins if they are required during development or to enhance functionality.
- The architecture must be designed with a user-friendly and easy-to-use user interface so users can easily perform the tasks like creating and managing groups and expenses.

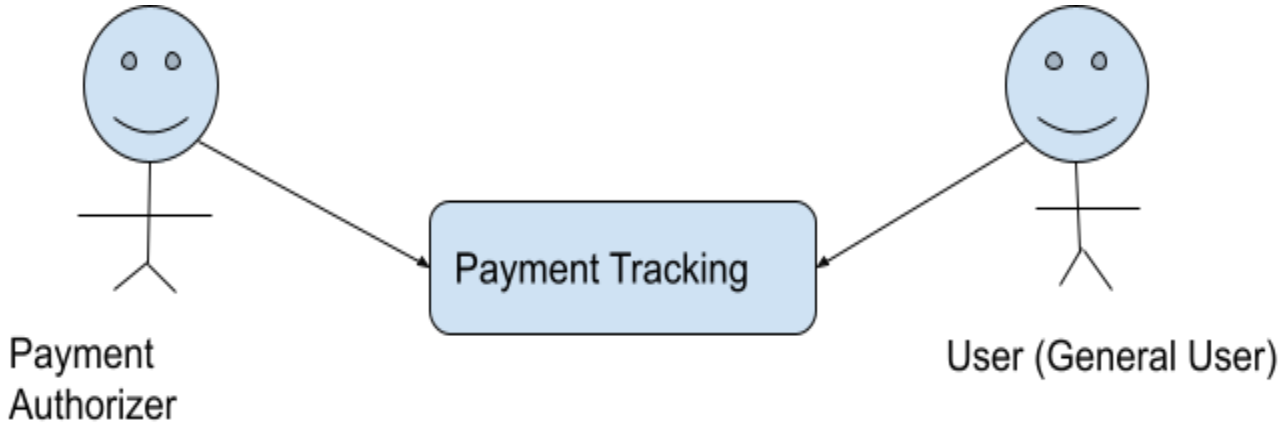
4) Use-Case View

Use cases for the split smart project are listed below

- Create Account
- Login to the Account
- Create/Manage Group

- Create New Expense
- New Expense Notification
- Balance Tracking
- Payment Tracking
- Report Receipt





4.1) Significant Use Case Descriptions

Create Account

This occurs when a user first time uses the app and creates a new account to use the system. The system stores the user details in the database allowing the user to use the system functionality.

Login to the Account

This use case occurs when a user login to the system using login credentials. The system matches the entered credential with the database and in case of correct information allows the user to enter the system dashboard.

Create/Manage Group

This use case occurs when a user creates a new group of participants to share and track the expense.

Create New Expense

This use case occurs when a user creates a new expense.

New Expense Notification

This use case occurs when the system notifies the users when any other group members create a group.

Balance Tracking
This use occurs when a user tracks the balance owed by other users.

Payment Tracking
This use case occurs when a user tracks payments that are made to settle the balances.

Report Receipt
This use occurs when the system provides users with reports of their expenses, balances, and payments.

5)Logical View

Overview

Presentation Package

- The presentation package provides a higher level view of the functionalities and classes in the form of boundary classes. These boundary classes correspond to lower level control classes in the application package

Application Package

- The application package is a lower level view of the implementation of the SplitSmart program. It provides a more detailed view of the classes in the form of control classes, which outline the major functionality of the program.

Domain Package

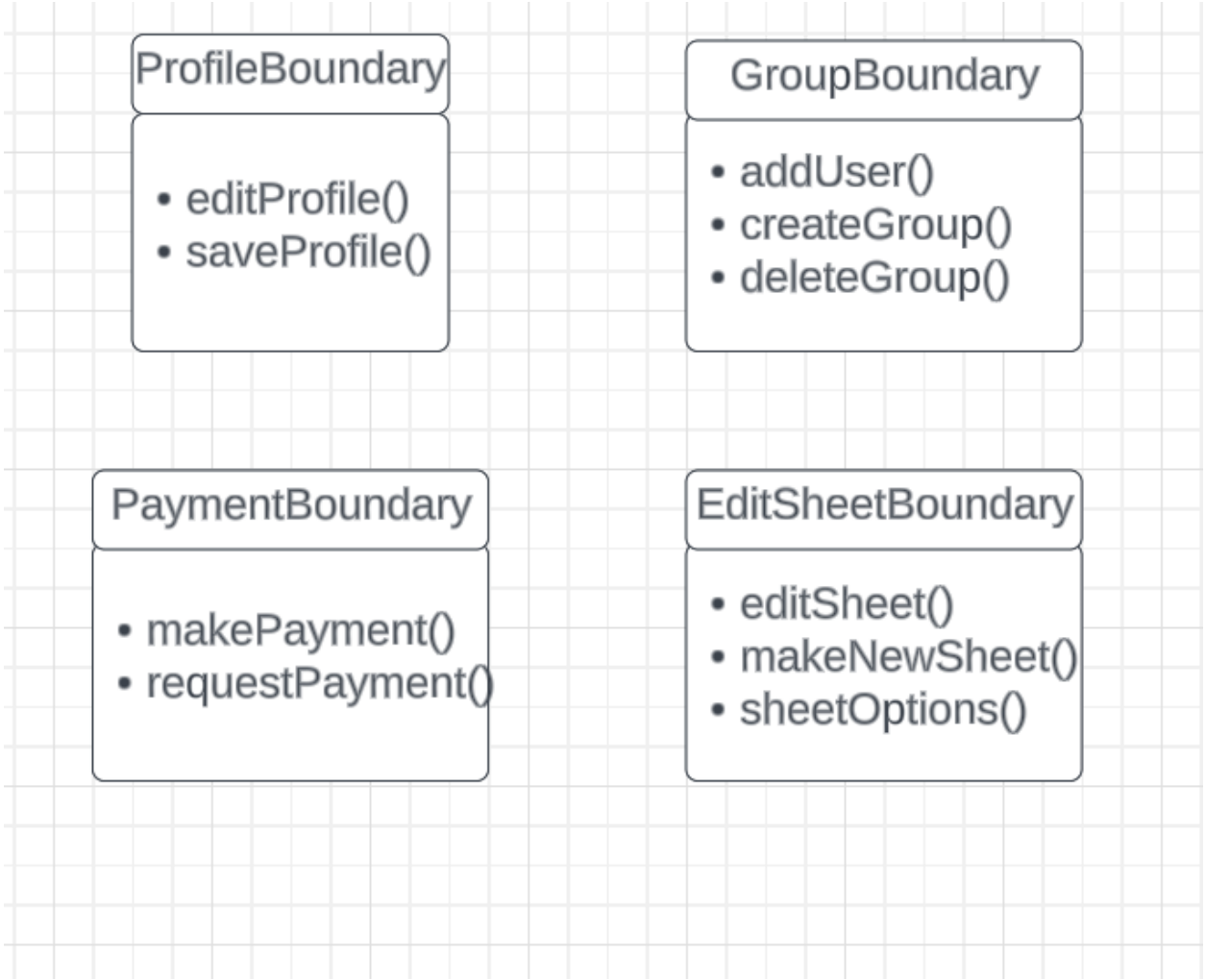
- The domain package provides a general overview of the system as a whole. The system is divided into subpackages.

Content Package

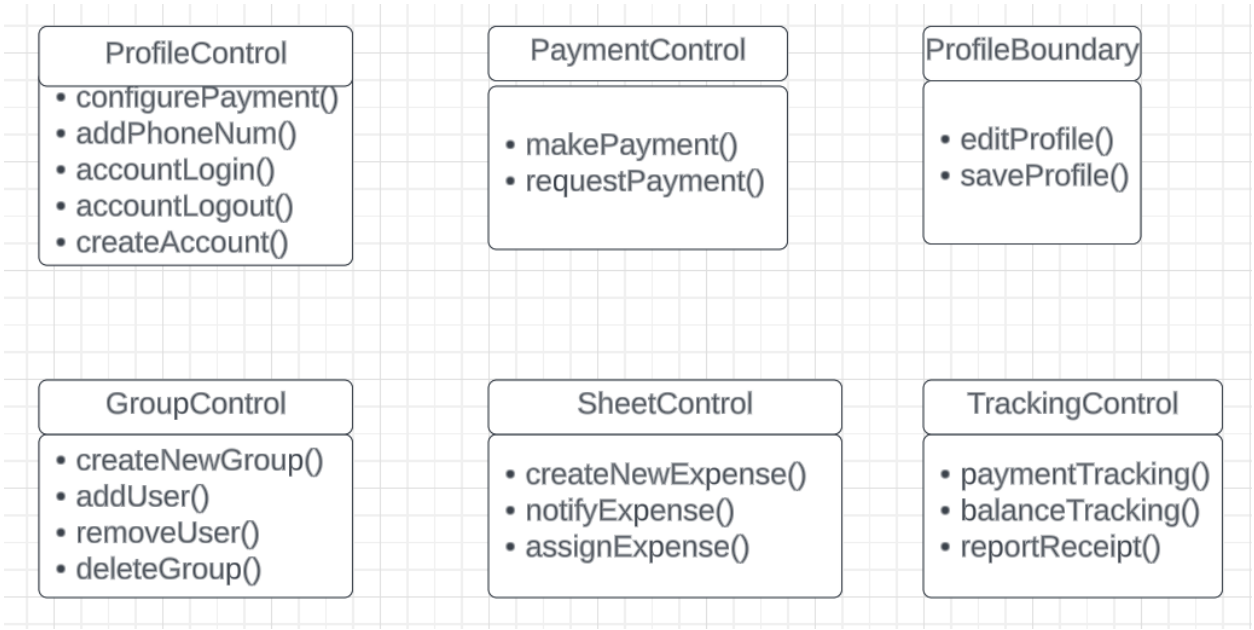
Profile Package

- The profile package is an overview of the profiles in the system. In the SplitSmart system there is only one user: the subscriber, or a user of the system. There can be admins for sheets but these are classified as normal users.

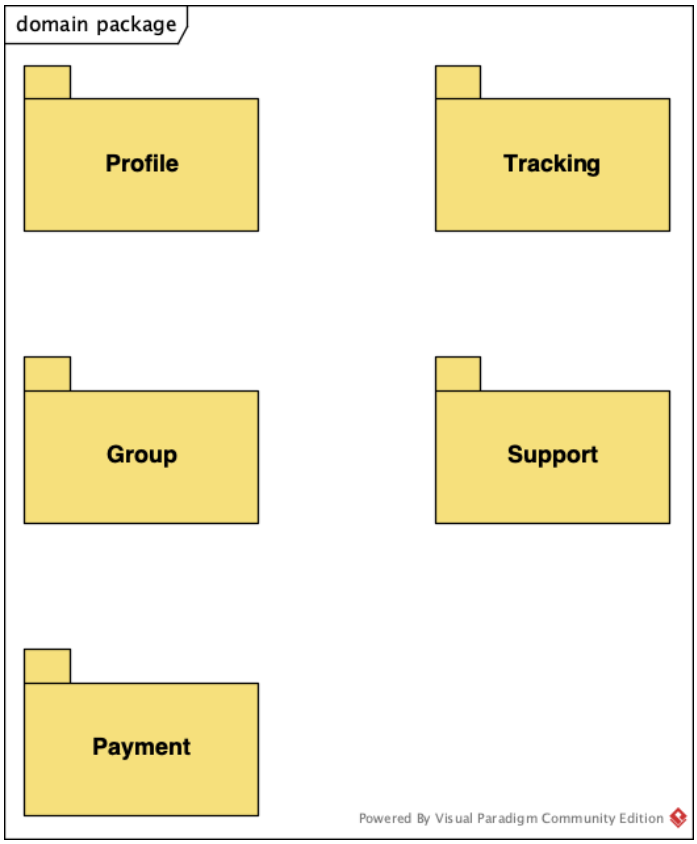
Presentation Package



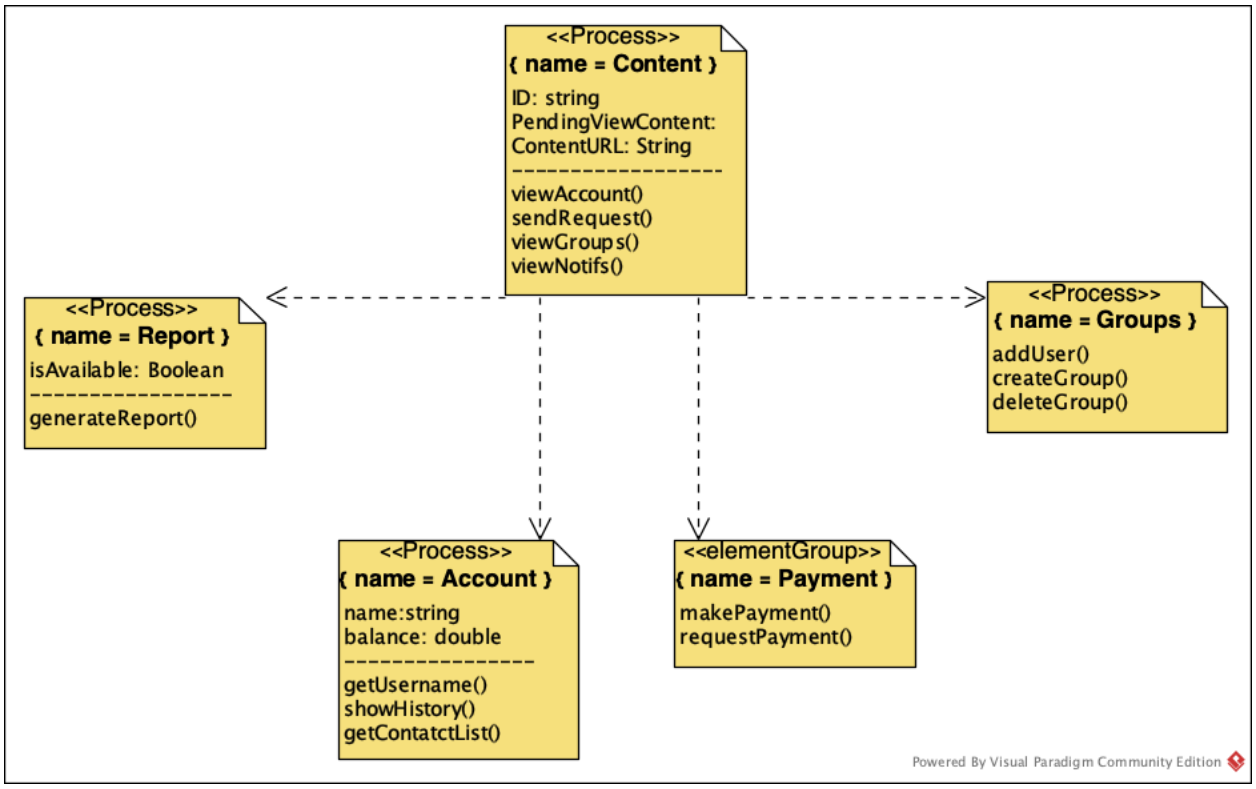
Application Package



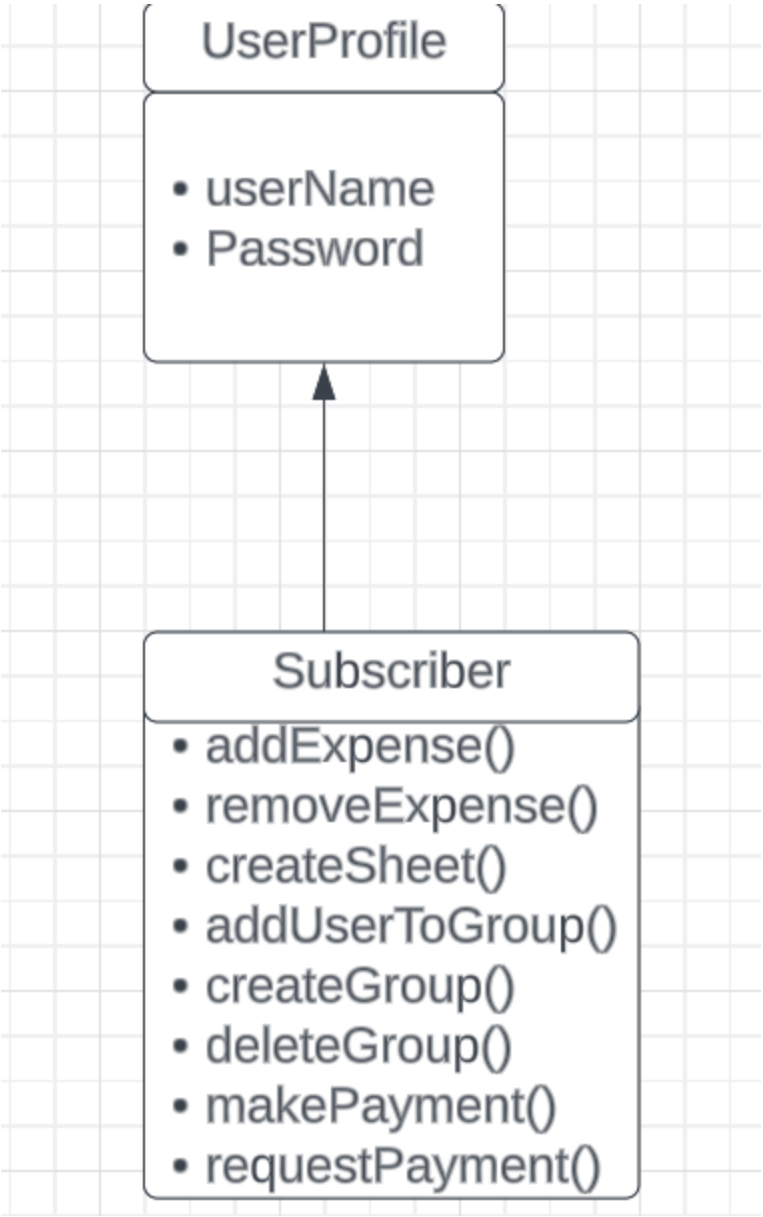
Domain package



Content Package

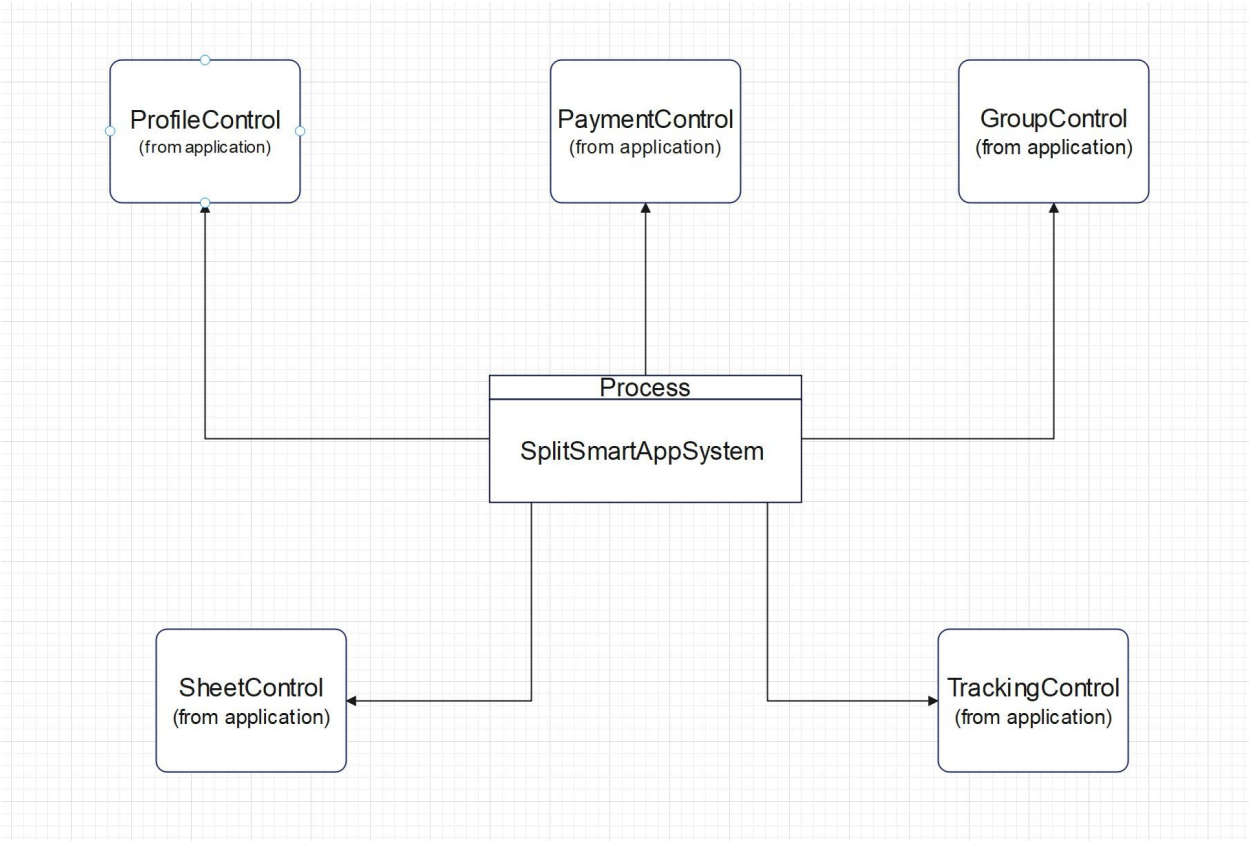


Profile Package



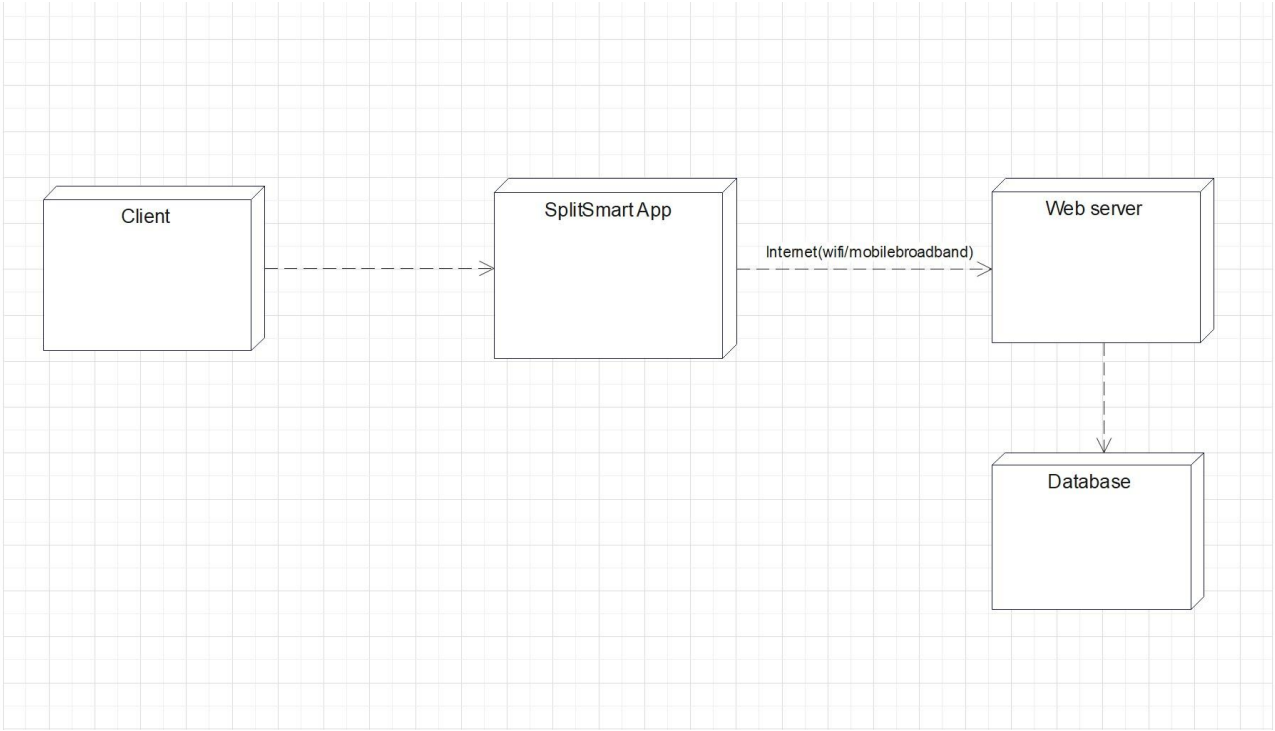
6) Process View

This diagram shows the single level processes that the server functions will be providing for the SplitSmart Application system.



7) Deployment View

The Client machine is any mobile device capable of downloading the SplitSmart app and has an internet connection able to connect to the SplitSmart Web server to be able to use the services the application offers.



8) Implementation View

The logical perspective, use-case view, process view, and deployment view are other architectural viewpoints. The implementation view's objective is to record the architectural choices taken for the implementation. describes the file system's organization of development artifacts. Files and folders make up the elements. Both deployment and development artifacts fall under this category. In a corporate setting, implementation may entail creating and then carrying out an innovative marketing plan to boost consumer product sales for the company.

9) Size and Performance

The projected quantity of valuable work completed in terms of time and resources. The amount of time required directly relates to how the code utilizes the CPU, resources utilized refer to computer memory, and the level of programming language expertise. The parity check matrix's dimensions are defined by its size. It depends on the language we choose as well as how many code lines there will be.

10) Quality

The code will be clean, straightforward, well tested, bug-free, documented, and effective. code that adheres to the criteria will vary depending on the demands of the business. It can be used on as many devices as are practical due to its mobility.