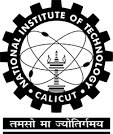
Software Requirements Specification for

7. NITC Split Bill System

**Version 1.0**

**Prepared by**

**Team Number: 8**

|  |  |  |
| --- | --- | --- |
| **Isha Gupta** | **M200694CA** |  |
| **Ayushi Kumari** | **M200687CA** |  |
| **Amarjeet Budhsaini**  **Aman Kumar Sankhwar** | **M200681CA**  **M200679CA** |  |
|  |  |  |
|  |  |  |

|  |  |
| --- | --- |
| **Project Owner:** | **Mr. KONU PRUTHVIRAJ** |
| **Course:** | **CS4096D Software Engineering Laboratory** |
| **Date:** | **24-01-2022**  **This template is based on the one available from the GMU site by Dr. Rob Pettit. Modifications specific to NITC are made and will be used for academic purposes only.** |

# 

# 2. Overall Description

## 2.1 Product Overview

The SplitPay project is a new, self-contained product intended for use on the Android platform.

While the SplitPay mobile application is the main focus of the project, there is also a server-side

component which will be responsible for database and synchronization services. The scope of

the project encompasses both server- and client-side functionalities, so both aspects are

covered in detail within this document. Below is a diagram of the SplitPay system which

illustrates the interactions between the server and client applications.

The SplitPay project is a new, self-contained product intended for use on the Android platform.

While the SplitPay mobile application is the main focus of the project, there is also a server-side

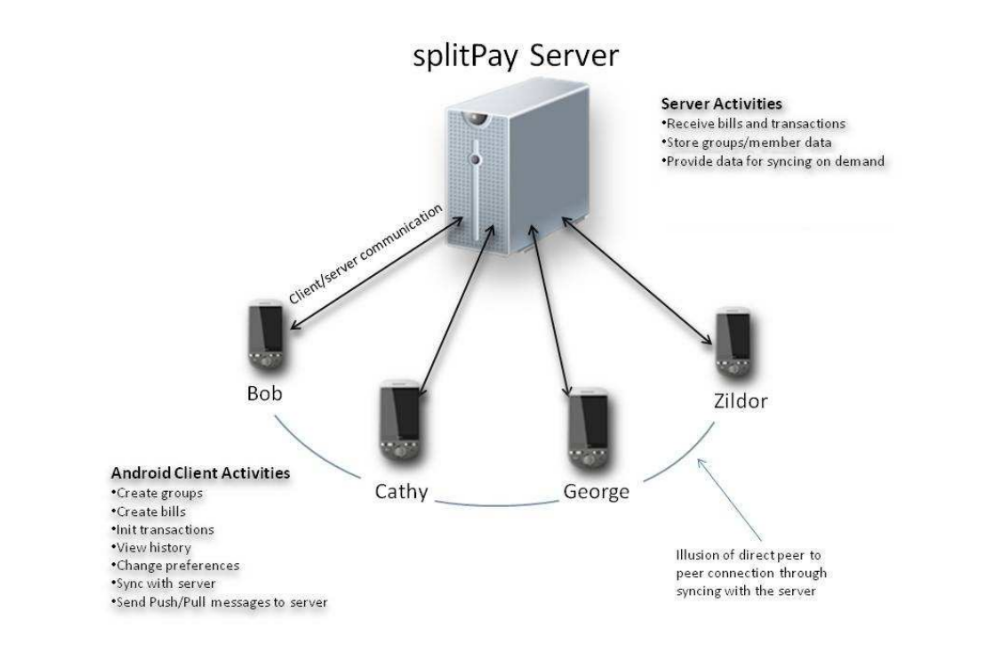
component which will be responsible for database and synchronization services. The scope of

the project encompasses both server- and client-side functionalities, so both aspects are

covered in detail within this document. Below is a diagram of the SplitPay system which

illustrates the interactions between the server and client applications.

The Project NITC Split Bill System is a new, self-contained product intended for use on the Android platform. While the NITC Bill mobile application is the main focus of the project there is also a server-side component which will be responsible for database and synchronization services. The scope of the project encompasses both server and client side functionalities, so both aspects are cover in detail within this document. Below is a diagram of the NITC Split Bill system which illustrate the interactions between the server and client applications.



## 2.2 Product Functionality

The following list offers a brief outline and description of the main features and functionalities of the NITC Split Bill System. The feature are split into two major categories:  
core feature and additional feature.  
core feature are essential to the application’s operation , whereas additional features simply add new functionalities. The latter features will only be implemented as time permits.

**Core Features**

1. User Registration & Welcome

Only appears once (the first time the application run)

Allows the user to register with the NITC Split Bill System server

Enables the user to customize his/her account settings and preferences

2. Group Creation & Management

Streamlines the process of creating and organizing groups

Provides support for multiple groups

Allows the user to add group members manually or from contact list

3. Posting a Bill

Stores and monitors the bill amount, the individuals involved, etc.

Includes support for multiple simultaneous bills

Efficiently distributes debt amongst the individuals responsible for the bill

4. Final Debt Resolution

Calculates the most efficient method of sorting out debts

Notifies group members of unresolved debts, credits, etc.

Offers the option to disband a group once all payment are made

5. Group History

Automatically records all transactions and bills posted to each group

Provides users with access to a detailed history of transactions

Supports sorting transactions by date, amount, payer, etc.

6. Show All Debts

Enumerates all of a user’s unresolved debts across each group he/she is a part of

Provides easy access to relevant information (past transactions, group info, etc.)

Offers the option to resolve a debt (or debts) immediately

7. Setting Menu

Allows the user to customize his/her preferences

Enables the user to modify certain features and functionalities

Can be accessed at any time using the built-in Settings button on Android phones

8. Help Menu

Displays a list of topics covering the different component of NITC Split Bill

Offers detailed information on each feature, menu, etc.

Can be accessed at any time via the settings menu

9. Push Notifications

Appear after any significant event occurs in a group

Alert group members of newly incurred expenses

Remind users of unresolved debts

**Additional Features**

10. Member Debt Visualization

Presents a visual representation of current member balances

Allows users to navigate through financial information in a more intuitive fashion

Automatically updates as users post expenses

11. E-mail/SMS Notifications

Extends the standard notifications service built into NITC Split Bill System

Automatically delivers notifications via e-mail and/or text message

Enables individuals without NITC Split Bill System to receive group notifications

12. NITC Split Bill Tutorial

Provides an abridged version of the Help menu for first-time users

Offers a step-by-step run through of each feature, menu, etc.

Enables any user to quickly and easily take advantage of all of NITC Split Bill System’s functionalities.

A major functionality present in several of these features is automatic synchronization. Using Android’s internet capabilities, the application periodically communicates with the NITC Split Bill server. This allows bills, transactions, groups, and group histories to be uploaded to a central server where the data can be shared with all other Android users in the group. This process of exchanging data between the server and the phone(s) is referred to as syncing.

For more detailed information, see Part 3 of the document (System Feature).

**2.3 User Classes and Characteristics**

The NITC Split Bill project is meant to offer a shared expenses solution that is faster, easier, and more convenient that manually calculating and handling debts. Consequently, the application will have little or no learning curve , and the user interface will be as intuitive as possible. Thus, technical expertise and Android experience should not be an issue. Instead, anticipated users can be defined by how they will use the product in a particular situation. The following list categorizes the scenarios in which NITC Split Bill is expected to be utilized:

**NITC Split Bill: Potential Scenarios**

1. Long-term recurring expenses(e.g. rent, groceries, utilities)

Key functions:

* Keep track of expenses
* Notify users when debts are incurred
* Record who has paid and who still owes

1. Short-term recurring expenses(e.g. travel costs – gas, food, hotel)

Key functions:

* Add new expenses (quickly and easily)
* Record who is paying and what he/she is paying for
* Update member balances on the fly

1. Single expense (e.g., splitting a bill at dinner)

Key functions:

* Create a group (quickly and easily)
* Add non-registered individuals to the group
* Quickly calculate each member’s balance

These groups are not meant to separate or categories users, just the different situations in which NITC Split Bill is likely to be used. In fact, a user may utilize the application for all of these scenarios simultaneously. This is another defining feature of the NITC Split Bill system: support for multiple groups. This functionality allows user to track expenses pertaining to several unrelated groups at the same time.

It is crucial that each of these situations be fully supported in the final product so as to maximize the overall value of the product. It is also important that the application be as user friendly as possible, otherwise it will not be a viable alternative to handling shared expenses manually. Most importantly, the application must be reliable. Regardless of the situation, the application must accurately distribute costs.

## 2.4 Design and Implementation Constraints

The primary design constraints is the mobile platform. Since the application is designated for mobile handsets, limited screen size and resolution will be a major design consideration. Creating a user interface which is both effective and easily navigable will pose a difficult challenge. Other constraints such as limited memory and processing power are also worth considering. NITC Split Bill is meant to be quick and responsive, even when dealing with large groups, so each feature must be designed and implemented with efficiency in mind.

## 2.5 Assumptions and Dependencies

Time Dependencies

As mentioned previously, the features of NITC Split Bill are divided into two groups: core features and additional features. Core features are crucial to the basic functionality of the NITC Split Bill application. These features must all be implemented in order for the application to be useful.

Optional features, however, are not critical to the function of the application. They are usability improvements and convenience, enhancements that may be added after the application has been developed. Thus, the implementation of these features is entirely dependent upon the time spent designing and implementing the core features. The final decision on whether or not to implement these features will be made during the later stages of the design phase.

Hardware Dependencies

Some of the additional features rely on hardware components present in Android handsets. For instance, the camera will be used to record images of receipts for digital storage. Consequently, this feature is entirely reliant upon the ability to access the camera’s functionalities.

External Dependencies

Several of the features presented in this document rely on the existence and maintained operation of several APIs. A non-exhaustive list follows.

**Email Notifications**

The Android platform is not suited for sending mass emails. Thus, the central server will be responsible for this feature of the application. The smartphone client will notify the server when message need to be sent using a custom API that is to be created. This API will use standard HTTP messaging to facilitate client-server communications. The API will be implemented using PHP.

**SMS Notifications**

This feasibility of this feature is yet to be determined. If implemented, this feature would allow offline users without an Android smartphone to receive notifications of outstanding debt and other information via text message. A suitable and free text messaging API that can be called from the server has yet to be found. The possibility of sending text messages from the Android smartphone client itself is also being reviewed.