Recepic – Budget Tracker

Version 1.0

A picture containing drawing

Description automatically generated

Eunbee Lee

Wooram Moon

Dohyun Kim

# **Table of Contents**

[Table of Contents i](#_Toc46518258)

[Revision History ii](#_Toc46518259)

[1 Introduction 1](#_Toc46518260)

[1.1 Purpose 1](#_Toc46518261)

[1.2 Document Conventions 1](#_Toc46518262)

[1.3 Intended Audience and Reading Suggestions 1](#_Toc46518263)

[1.4 Project Scope 1](#_Toc46518264)

[1.5 References 2](#_Toc46518265)

[2 Overall Description 3](#_Toc46518266)

[2.1 Product Perspective 3](#_Toc46518267)

[2.2 Product Features (Functions) 3](#_Toc46518268)

[2.3 User classes and Characteristics 4](#_Toc46518269)

[2.4 Operating Environment 4](#_Toc46518270)

[2.5 Design and Implementation Constraints 5](#_Toc46518271)

[2.6 User Documentation 5](#_Toc46518272)

[2.7 Assumptions and Dependencies 5](#_Toc46518273)

[3 External Interface Requirements 6](#_Toc46518274)

[3.1 User Interfaces 6](#_Toc46518275)

[3.2 Hardware Interfaces 6](#_Toc46518276)

[3.3 Software Interfaces 6](#_Toc46518277)

[3.4 Use Cases 7](#_Toc46518278)

[4 Functional Requirements 13](#_Toc46518279)

[4.1 Formal Use Case Description 16](#_Toc46518280)

[4.2 Use Case Diagram 18](#_Toc46518281)

[5 Nonfunctional Requirements 19](#_Toc46518282)

[Appendix A Glossary 20](#_Toc46518283)

[Appendix B Analysis Models 21](#_Toc46518284)

[Appendix C Stakeholder Register 22](#_Toc46518285)

[Appendix D Interview Questions 23](#_Toc46518286)

[Appendix E Object Oriented Modelling 25](#_Toc46518287)

[E.1 Class-Responsibility-Collaboration Cards 25](#_Toc46518288)

[E.2 Domain Class Diagram 29](#_Toc46518289)

# **Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Date | Reason for Changes | Version |
| Eunbee Lee  Wooram Moon  Dohyun Kim | Jun 19, 2020 | Summer 2020 Assignment – Part A | 1.0 |
| Eunbee Lee  Wooram Moon  Dohyun Kim | Jul 24, 2020 | Summer 2020 Assignment – Part B | 1.1 |

# **Introduction**

## Purpos**e**

This new software will provide tools for managing the budget, primarily for but not limited to personal needs.

## **Document Conventions**

### Acronyms

|  |  |
| --- | --- |
| Acronyms | Description |
| DB | Database |
| SSR | Server-side rendering |
| SQL | Structured query language |
| API | Application programming interface |

## **Intended Audience and Reading Suggestions**

This document is intended to provide information on this software for the following groups of people.

1. Developers and project managers
2. Documentation writers
3. Users, both personal and business

## **Project Scope**

There are many services that provide budget managing, but most of them are focused on managing only personal finance. But in real life, people often face situations where they need to manage shared budgets, such as planning and tracking household budgets or sharing costs to go on a trip with friends. This software provides a group budget managing feature with access control so that the users can freely create groups, set who can edit the data including budget plans and transaction history, and delete the groups with the balance deposited into the personal budget records of the members as configured.

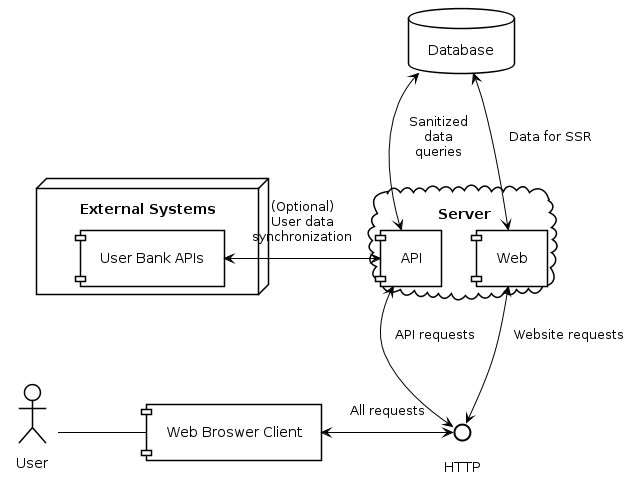
As a budget tracker, all features provided for group budgeting are also provided for personal uses. The features include adding, editing, and removing account and card information, managing transaction records, setting weekly, monthly, or yearly budgets, and tracking and showing the income and expenditure with category and tag information.

## **References**

# Overall Description

## **Product Perspective**

The new software is a new competitor among budget tracking applications. It is a self-contained product with its own standalone solutions. It can integrate or communicate with external banking systems to provide convenience features such as automatic insertion of transaction records and account and card information synchronization.



## **Product Features (Functions)**

The new software will provide the following functionalities:

1. Allow each user to keep their own budget plan and records
2. Show statistics of the budget management status
3. Make groups to manage shared funds
4. Synchronize various information from users’ banks, including accounts, cards, and transaction records

## **User classes and Characteristics**

### Family

The advantage is that it is easy to use and simple. It consists of five categories: expenditure history, income history, budget management, data statistics, and data charts. You can specify the start date of the month according to the payment date of the card, making it easy to settle monthly expenses.

If you used a card installment, you enter the installment amount in installments. Particularly, when canceling the card approval history already paid, the record is automatically deleted by recognizing the cancellation text sent from the card company. It is a household account app that can write household accounts as smart as its name.

### Bank or Companies

Interlocking with major banks, companies and credit card accounts allows for integrated inquiry of bank information at once. It is a household account app that does not need to organize the deposit and withdrawal history separately by automatically importing all the details such as income, expenses, and account transfer. In particular, by analyzing your consumption pattern information, you can also receive a personalized card that can give you the greatest benefit among 3,600 cards. It also has the advantage of being able to look up credit ratings for free, as well as to check real estate and automobile prices.

## **Operating Environment**

### Server

The web and API server will

1. Run on a Unix server
2. Talk to PostgreSQL server, which is used to store information
3. Be developed using NodeJS
4. Support SSR

### Client

The client application will

1. Run on the following browsers: Chrome, Firefox, Safari, Opera, and Microsoft Edge
2. Utilize service worker to provide app-like experience
3. Support using camera on mobile devices

## Design and Implementation Constraints

## User Documentation

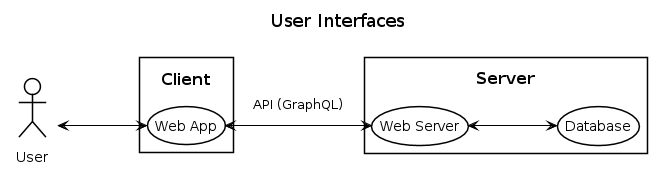
## **Assumptions and Dependencies**

#### Integration with Services from Banks

It is assumed that it is possible to use some API provided by the banks that the users use. It might be possible that the banks do not provide such API or that they refuse to allow the new software be integrated with their service at the first place.

# External Interface Requirements

## **User Interfaces**



The users will use web browsers from any device to get access to the client. The targeted browsers are the major modern browsers: Chrome, Firefox, Safari, Opera, and Microsoft Edge. The application logic will be mostly in javascript with some performance critical parts ported to WebAssembly.

The server and the client will communicate using GraphQL protocol. The server is run as a web server using NodeJS to utilize SSR, packed into Docker containers. There will be a server instance dedicated to hosting the SQL database for the web server.

## **Hardware Interfaces**

### Desktop and Mobile Devices

Any devices that have web browsers can use the web browsers to access the service.

### Camera

Cameras from mobiles devices are used to scan cards and receipts. This enables fast and convenient data insertion for the end users.

## **Software Interfaces**

### Web Browser UI Library

Applications with moderate to high complexities that run on web browsers can be developed with much less pain using UI libraries, both for programming and styling.

### GraphQL Implementations

The client and server need GraphQL protocol implementations to use the protocol when communicating with each other.

### Text Recognition

The software needs to be able to recognize texts in the pictures provided by the users in order to parse the data and add the information to its database.

## Use Cases

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Requirements | Actors | Description |
| Create a user account | FR01 | User | The actor will click or touch on the create account button. The software will present a form for user account details like email as a response. After the actor fills in all the required fields and submits the form, the software will create the account for the user and send a link for account activation to the provided email address. |
| Activate a user account | FR01  NFR06 | User | The actor will navigate to the user account activation link provided in the email sent to them in the process of creating the account. The software will confirm that it is a valid link and is accessed within a specified amount of time. Then, the software will enable the user account to access all of its services by marking the account as valid and activated. |
| Insert bank account information | FR05 | User | The actor will click on the button for adding a new bank account. The software will present a form with necessary fields for the information. After the actor fills in the form and submits, the software will create an entry for the actor’s bank account information. |
| Remove bank account information | FR05 | User | The actor will click on the button to remove a bank account they previously added. The software will show a confirmation dialog to make sure the actor wants the removal to be performed. After the actor confirms it, the software will remove the bank account information and all other information, including card information and transaction entries, related to the bank account. |
| Insert card information | FR05 | User | The actor will click on the add card button associated with one of their bank account entries. The software will present a form to the actor to have them fill in the fields for necessary information. After the actor submits the form, the software will create an entry of the card information for the actor. |
| Remove card information | FR05 | User | The actor will click on the remove card button, and the software will present a dialog to make sure the actor wants the removal. After the actor confirms the action, the software will remove the card information and its associated information such as transaction entries. |
| Change card information | FR05 | User | The actor will click on the button for changing information on a card entry. The software will present a form with the previously entered data as the values. After the actor makes desired changes and confirms to save the information, the software will save the new information. |
| Insert transaction information | FR05  FR08 | User | The actor will click on the insert transaction button for one of their bank accounts or cards or for one of the dates in the calendar. The software will present the actor with a form to fill in about the transaction to be inserted. After the actor submits the form, the software will add a transaction entry with the supplied data. |
| Insert transaction information from a receipt | FR05  FR07 | User | The actor will click on the insert transactions from the receipt button for one of their bank accounts or cards. The software will try to gain access to the camera of the actor’s device so they can take a picture of the receipt. After the picture is taken, the software will process the picture and read in the data, and then it will present a form with the parsed data so that the actor can make any necessary changes. Once the actor is satisfied with the transactions data, they will submit the form, and the software will add the entries. |
| View inserted transactions | FR05 | User | The actor will click on the button to view their transaction records. The software will present a list of inserted transaction entries sorted by date. There will also be options to sort and filter the entries by date, category, and the range of the amount of the transaction. |
| Change inserted transactions | FR05 | User | The actor will click on the modify button for a transaction entry in interest. The software will then present a form with fields filled in with the previous data, ready to be modified. The actor will make any changes to the data and submit the form. The software will save the newly submitted data and return to the view transactions page. |
| Remove inserted transactions | FR05 | User | The actor will click on the button to delete entry of the transaction record to delete. The software will prompt the actor to confirm the removal of the record. If the actor clicks the cancel button, the prompt exists and nothing will be changed. If the actor clicks on the confirm button, the selected transaction entry will be removed from the database. |
| Create a group | FR04 | User | The actor will click the ‘Managing group’ button under the menu button. The software will present a ‘Managing group’ page where the actor can manage the group creation, group removal, user deportation, and user permissions. This page will contain the following data input features that are related to creation of a group: Name of the group, Number of users(2 is default), and password for the group(optional). After the actor fills up the data, the actor will click the ‘submit’ button to create the group. Once the group is created, the page will be automatically closed and connected to the ‘Invite’ page. |
| Remove a group | FR04 | User | The actor will click the ‘Managing group’ button under the menu button. The software will present a ‘Managing group’ page where the actor can manage the group removal, group creation, user deportation, and user permissions. This page will contain the following data input features that are related to removal of an existing group: group selection, and notice comment area(Actor can send a message to group members to inform the removal of the group and it is optional). After the actor fills up the data, the actor will click the ‘submit’ button to remove the group. Once the group is removed, the page will be automatically closed and returned to the main page. |
| Expel a user from a group | FR04 | User | The actor will click the ‘Managing group’ button under the menu button. The software will present a ‘Managing group’ page where the actor can manage the user deportation, group removal, group creation, and user permissions. This page will contain the following data input features that are related to deportation of the group member: group selection, member selection and notice comment area(Actor can send a message to group member to inform his/her expulsion and it is optional). After the actor fills up the data, the actor will click the ‘submit’ button to remove the member. Once the member is expelled, the page will be automatically closed and returned to the main page. |
| Invite other users to a group | FR04 | User | The actor will click on the ‘Invite’ button under the menu button. The software will present a ‘Invite’ page where the actor can send an invitation to another user. This page will have the following data input area: ID of another user or e-mail of another user, password, and invitation comment. After the actor fills up the data, the actor will click the ‘submit’ button to send the invitation. Once the invitation has been sent, ‘invite’ page will be automatically closed and returned to the main page. |
| Change user permissions in a group | FR04 | User | The actor will click the ‘Managing group’ button under the menu button. The software will present a ‘Managing group’ page where the actor can manage the user permissions, group removal, group creation, and user deportation. This page will contain the following data input features that are related to changing user permissions of a group: group selection, member selection, authority selection(check box with following options: Read only, Read & Write, Data Download, Expenditure analysis, Inviting new user, expulsion of existing group member, changing user permission, All). After the actor fills up the data, the actor will click the ‘submit’ button to set new user permissions for the group. Once the user permissions are set, the page will be automatically closed and returned to the main page. |
| Request expenditure analysis | FR09 | User | The actor will click on the ‘Analysis’ button under the menu button. The software will present the actor with an analysis about the expenditures that user has made. After the actor looks through the analysis, the actor will click the ‘back’ button to return to the main page. |
| Request download of data | FR10 | User | The actor will click on the ‘Download’ button under the menu button. The software will present the actor with a form to fill in about the transaction that the actor wants to download. Form should be filled up with the following data: range of dates/weeks/months, type of expenditures. After the actor submits the form, the software will download .xls file with the selected data. |
| Connect to external bank services | FR12 | User | The actor will click on the button to connect records to an external bank service. The software will present a page where the actor can select which bank they wish to connect their account to among the ones that are supported. After the actor clicks on the one they want, the software will present a page where the actor can enter details of their accounts for the selected bank service. This page can be either served from the software or from the bank itself. After the actor is successfully authorized by the external bank service, the software presents a page that informs them the connection has been established. |

# Functional Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Title | Description | Priority | Requester |
| FR01 | Create an account | Each user can create an account. | High | Investor |
| FR02 | Create and remove groups | User is able to create a group for a shared fund. The group can have its own accounts and cards. The accounts can be real banking accounts or virtual ones for managing the funds. The user is also able to remove any groups they are the owner of. | High | User |
| FR03 | Inviting other users to groups | Inviting other users to share a group to managing service. This function will be conducted through pressing Invite button.  Following features will be shown when a user clicks the Invite button:   1. Text input for entering ID of the user to be invited 2. Text input for password for the shared data 3. Button that sends the invitation | High | User |
| FR04 | Authority permission | The main user of a group can decide authority of co-users who were invited to the group.   * Read only: co-user can see the data and expenditure analysis and download the data, but they cannot write any data in the group’s account. * Read and Write: co-user can use all of the functions in the group’s account except for inviting another user and removing the account. * Full authority: co-user can use all of the functions in group’s account including inviting and removing other users. | High | User |
| FR05 | Data input | User can input the data of the following:   1. Date 2. Income 3. Expenditure 4. Type of expenditure (ex: shopping, groceries, travel, education, transportation, etc.) 5. Type of income (Salary, Investment, etc.) 6. Path of expenditure (cash, credit card, debit card) 7. Amount of expenditure/income 8. Memo | High | Investor |
| FR06 | Internet Connection | User can use the application through any device of choice by connecting to the server using web browsers. | High | Investor |
| FR07 | Camera Connection | User can take images of their receipts. Added to this, user can store and retrieve images in their accounts. Images will be stored under certain date that they chose. If the user does not choose a particular date, the image will be automatically stored with the date the picture was taken. | Medium | User |
| FR08 | Calendar Connection | Calendar is necessary for the user to have daily basis bookkeeping system. Once the user clicks the date in calendar, data input box will be opened, and the data will be inserted into the user’s account once confirmed. | High | User |
| FR09 | Expenditure analysis | Program automatically analyzes the expenditure of the user by each category. The analysis contains total expenditure and income, expenditure per category, and the user’s spending habits. | High | User |
| FR10 | Download | Bookkeeping data can be downloaded as .xlsx file. The user can choose the fields they want to include.  Ex) cash expenditure only, shopping expenditure only | Medium | Investor |
| FR11 | Advertising | Creating banners that can be seen by users, so investor can post advertisement in the banners to generate income. | High | Investor |
| FR12 | Bank connection | Connect the program with bank application, so that expenses made through credit card or debit card will recorded in the user’s account automatically. | Low | User |
| FR13 | Chat service | Users can chat with co-users in the same group. | Low | Investor |
| FR14 | FAQ message service | Users can ask questions about the application via sending message. | Medium | User |

## Formal Use Case Description

### Insert Card Information

#### Interaction

Card using.

#### Primary Actor

Customers.

#### Goal in Context

How to insert card information into the application database.

#### Preconditions

System must be configured and user information or cards must be obtained.

#### Trigger

The customers have to insert their card information.

#### Scenario

1. Customers decide to choose the application.
2. Customers download apps for creating the card
3. Customers need a Id.
4. Customers enter their password (each 8 digits and add characters, uppercase, at least one number)
5. Card inserts and check app for working.
6. The app also supports the card to show the money balance where customer used.

#### Exception

1. ID or password digits are incorrect or not recognized.
2. Customer selects the apps, which customer use easily.
3. The app is useful to check easily for customers.

#### Priority

Top priority.

#### Secondary actors

System administrator, Database manager.

#### Channels to secondary actors

1. System administrator: It has been protected customer information by hackers.
2. Data base manager: Customers information data.

#### Open issues

1. What apps are more safety for customers?
2. Are there any critical issues for housekeeping apps?

## Use Case Diagram

# Nonfunctional Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Title | Description | Priority | Requester |
| NFR01 | Back-up | System has to be backed-up automatically on a weekly basis. | High | IT Security officer |
| NFR02 | Security - Cyber | Following things should be conducted:   1. Penetration testing 2. Load testing 3. Security auditing 4. Vulnerability scanning 5. Security scanning | High | IT Security officer |
| NFR03 | Response time | Maintain the response time below 2 seconds. | High | IT department chief |
| NFR04 | Informing message | In case the system has to be updated, system will produce message to inform to users. | Low | IT department chief |
| NFR05 | Web browser compatibility | The system should be compatible with the following browsers: Firefox, Chrome, Microsoft Edge, Safari, and Opera. | High | IT department chief |
| NFR06 | Security - Authentication | The following features should be provided:   1. Logging in 2. Finding lost password | High | Investor |

# Glossary

# Analysis Models

# Stakeholder Register

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Name | Stakeholder  Position | External/Internal | Stakeholder contact details | Operational/ Executive | Interest  (high, medium, low) |
| Hailey Duke | Accountant (Advisor) | External | 3hihailey@gmail.com | Operational | Medium |
| Tily Rim | Investment specialist (Advisor) | External | tilyrim@gmail.com | Operational | Medium |
| Bella Burton | Investor | External | bburton65@gmail.com | Executive | High |
| Jake Gregory | IT Security officer | Internal | jg770922@bigit.com | Executive | Low |
| Mina Jang | User | External | mmmjang@naver.com | Operational | High |
| Kevin Chang | IT department chief | Internal | kckkk@bigit.com | Executive | Medium |
| Gloria Shin | Marketing officer | Internal | g91shin@yahoo.com | Operational | High |
| Carl Louis | Financial manager | Internal | carllouiscarl@gmail.com | Executive | Medium |
| Dan Lee | Programmer | Internal | dandanxoxo@bigit.com | Operational | High |
| Sarah Xing | Programmer | Internal | sarahsara1@ bigit.com | Operational | High |
| Emma York | Program tester | Internal | emnewyork@bigit.com | Operational | High |

# Interview Questions

|  |  |  |
| --- | --- | --- |
| Question | Stakeholder position | Answer |
| 1. Is it better to create balance sheet & income statement? | Accountant (Advisor) | Since this is not the program for the tax return or annual report, no need to make those functions. However, abridged version of income statement is recommended feature for the program. |
| 1. What is abridged version of income statement? | Accountant (Advisor) | Statement that include income, sum of income, expenditure of each category, sum of expenditure, and net income. You can display net income either + or -. (+ : consumption is less than the income; - : consumption is greater than the income) |
| 1. Which feature do you need the most? | User | Expenditure analysis by per categories |
| 1. What type of people will use this application? Professionals? | Marketing officer | Not the professionals, normal people who does not know accounting |
| 1. What kind of equations will be needed to provide expenditure analysis to users? | Investment specialist (Advisor) | Simple mathematics will be needed. Just sum up all the expenditures by per categories and compare the sum of expenditure to budget & income. |
| 1. Which language will be used in this service? | Investor | English & French |
| 1. Who should I talk to if my bank card is not working? | IT security officer | IT security officers help customers with security-related problem of bank information or others. |
| 1. What type of Database is used in your company? | Database Administrator | Oracle |
| 1. What do IT people work for in this company? | IT administrator, Programmer | Develop services, platforms, and developments needed by customers. |
| 1. Why do you need a household account app? | User | As the card usage history is automatically entered into the household account app, customers can save money. |
| 1. What is the ****financial technology**** of the household account app in your company? | Investor | Simple household app for customers and bank household for other companies. |

# Object Oriented Modelling

## Class-Responsibility-Collaboration Cards

|  |  |
| --- | --- |
| Transaction | |
| Card information | Card |
| Bank account information | Bank Account |
| Date |  |
| Category |  |
| Amount |  |
| Modify transaction |  |
| Receive information from records tracker | Records Tracker |

|  |  |
| --- | --- |
| Expenditure Analysis | |
| Requester | Account |
| Receive transaction information | Group |
| Analyze the expenditure |  |
| Create .xls file |  |

|  |  |
| --- | --- |
| Card | |
| Owner | Account |
| Group | Group |
| Bank account | Bank Account |
| Card number |  |
| Card usage |  |
| Modify card information |  |

|  |  |
| --- | --- |
| Bank Account | |
| Bank |  |
| Owner | Account |
| Transaction information | Transaction |
| Sync information from external bank service | Bank |

|  |  |
| --- | --- |
| Records Tracker | |
| Receive receipt reader information | Receipt Reader |
| Receive card information | Card |
| Send card information | Card |
| Receive bank account information | Bank Account |
| Send bank account information | Bank Account |

|  |  |
| --- | --- |
| Bank | |
| Send bank account information | Bank Accout |

|  |  |
| --- | --- |
| Receipt Reader | |
| Read receipt | Camera |

|  |  |
| --- | --- |
| Camera | |
| Take picture of receipt |  |
| Store the picture |  |

|  |  |
| --- | --- |
| Account | |
| Username |  |
| Password |  |
| Email |  |

|  |  |
| --- | --- |
| Permissions Manager | |
| Give authority | Group, Account, Permissions |
| Remove authority | Group, Account, Permissions |
| Expel member | Group, Account, Permissions |

|  |  |
| --- | --- |
| Permissions | |
| User | Account |
| Authority |  |

|  |  |
| --- | --- |
| Group | |
| Owner | Account |
| Members | Account |
| Permissions for each member | Permissions |
| Create group |  |
| Remove group |  |

|  |  |
| --- | --- |
| Invitation | |
| Sender | Account |
| Receiver | Account |
| Send invitation |  |
| Accept invitation |  |

## Domain Class Diagram

