SW Engineering CSC 648-04

“Track Your Money (TYM)”

Team 6

10 / 3 / 2022

Team members:

Robert Sato (Team Lead),

Siham Argaw (Scrum Master),

Hemanta Thapa (Front-end Lead),

Neel Manthani (Back-end Lead),

Diego Islas (Github Master)

1. Execution Summary

A common problem people experience is trying to control their spending. Much of the difficulty arises because people don’t know exactly where their money is going. Because of this, when they make efforts to lower their expenses, they don’t know if their actions are making a real difference, and this lack of feedback makes it difficult to continue with any positive habits that they may have tried to adopt.

This is where the Track Your Money app comes in. With this app, users will be able to track their expenses quickly and easily, letting the app be their own personal ledger. The app will provide the users with categories, some premade and others user-created, that will allow the user to see what exactly they are spending most of their money on. Using this feature, the user will be able to cut out unnecessary spending with greater precision without having to spend the time tallying up each purchase themselves. After using the app for a certain period of time, users will be able to see graphs tracking the changes in their spending over the course of months or years. This will allow them to notice unwanted changes in their spending, and adjust quickly to remedy it. This app will also allow users to input their income, and will calculate the net gain or loss of money. This, along with a feature that allows the user to set a budget for themselves, will help users make sure that they are saving an amount of money that they are happy with.

The team behind this application are all college students who believe in helping individuals take control of their spending. Being in college, we are acutely aware of the rising costs of living and other services, and understand the importance of making sure money goes to the right places. We want to use this app to provide users with greater peace of mind in regards to their spending and empower them to use their money in ways that they can be satisfied with.

1. Personas and User stories
   1. Persona 1
      1. CEO

**Jordan Smith**

| **Info:** 40, CEO  **Bio:** Jordan travels around the US often in order to conduct business. He often racks up travel expenses due to flights and hotel stays. He has very little time to count up the expenses because he’s extremely busy with work. | **Wants & Needs:**   * A quick way to track expenses * A way to categorize business expenses so they can be deducted from taxes   **Frustrations:**   * Doesn’t want to have to remember every expense throughout the day * Concerned that unnecessary spending may be costing him profits |
| --- | --- |

* + - 1. The way Jordan will use the application is to monitor where their money is going, making sure to include what is business expense and what is personal expense on their tax returns to increase their deductible. Given the option to see what is deductible can make it easier to keep track
  1. Persona 2
     1. College Student

**James Rowe**

**Info:** 22, Criminal Justice Major

**Bio:**

Lives in a dorm, one of the school's dorms in Sacramento State University and is sharing with three more other students that are attending there as well. James likes to go out and spend his time with his roommates outside of his class hours. He is also a hard working student that is studying to become a lawyer but manages his time wisely.

**Wants & Needs:**

* Be able to manage their money in an organized manner, making it easy to keep track and see where all transactions
* Be able to write down expenses without any extra steps

**Frustration:**

* Does not want to spend money on any application that can do the same thing he can on a piece of paper.
* Feels as though he is spending to much on going out and would like to see why he is barely able to pay rent for the dorm
  + - 1. Although James might not be able to write his expenses down on a piece of paper every time, they would be able to put it on their phone since they are always carrying it around. Knowing that college expenses can cost a lot in today’s society, James will be able to note down what to expect if they would pay for the classes for a given semester and this will show him how much they would have and give a better insight on how to manage their money for other necessities.
  1. Persona 3
     1. Restaurant Worker

**Hattie Barajas**

**Info:** 29, Cafe Host

**Bio:**

Is a hard worker at a cafe, living in an apartment in San Francisco with her cat. She enjoys buying treats for her cat, going out for her morning jogs and likes to read during the day. She has a roommate that she gets along with, going out to museums and the beach during the summer time.

**Wants & Needs:**

* Be able to keep track of cash tips rather then putting all of it into their bank account
* Wants to save up to be able to live in a bigger places to have more room

**Frustration:**

* Does not like the feeling of living in such a small compact place and having to pay so much for it
* Finding out that she had less money then what they have expected due to not being able to see how much cash was left
  + - 1. Hattie will be able to use the application and be able to keep track of tips especially since there is no actual log on how much cash they actually have. This will give her a rough estimate on how much cash you have and can be a better way to manage only your cash and/or even card transactions since everything will be logged. Hattie will be able to view graphs from previous months and weeks and click on the given month or week and will display the transactions.

1. Data Definitions

The main data structures or collections are going to be users, expenses, and income. Each user will have a username, password, email, and id. Transactions will be a base interface class. It will keep track of data such as date, amount. Then we will have a subclass with expenses and incomes as transactions. They will have separate data fields that are exclusive to their own type. Expenses will have data fields of category of expense, description and everything inherited from transactions. Income will have the data fields of type of method of deposit and everything inherited from the transaction class. We will be using a nosql database so there are no relations. Each transaction will be its own document. And each user will be its own document. Transactions can be an expense or an income. We can also have data fields that would belong to the users collection that track last time paid for subscription and when the subscription payment is coming up.

Users will have the privilege to see user specific information based on the login information they provide. They will only be able to access the information when logged in. They will be able to see that accounts expense and income data fields

The income and expenses data fields will be only user specific. Each expense or income (transaction) will have an id that corresponds to a userid. So only the user that owns those transactions can see their income/expense related data.

Users

. username

. email

. password

. id

Income

. id

. Date

. Type of method of deposit

. Amount

Expense

. id

. Date

. category

. description

. Amount

1. Initial list of functional requirements
2. Priority 1
3. Users can add spendings manually
4. Users can add their income manually
5. App keeps a record of spendings and income
6. App calculates savings per month and year and displays it
7. App categorizes spending by type, date spent, amount spent, and description, and these categories can be edited by the user.
8. App categorizes income by paycheck, tip, direct deposit, and cash and these categories can be edited by the user.
9. Users can delete an expense or income they added
10. Users have to login/ sign up to access app
11. Priority 2
12. Allows users to input a certain budget that they want to set for themselves per month and displays a message when they pass that limit.
13. App displays graph of income and expenses throughout a month and a year
14. Priority 3
15. App lets users add images of receipts to their spending, and it can be used for reference.
16. List of non-functional requirements
17. App is compatible with mobile browsers and all desktop browsers
18. Data is stored in Mongodb database on Google Cloud Server
19. App provides strong password encryption
20. App is easy to use, users can easily add their spending and income to the app, and the app will add it to the record and provide them with their savings.
21. Users will get notified when they have passed a certain spending limit, which they have set for themselves, through a message, which they can easily access.
22. We will take advantage of the mongodb replica set. And also MongoDB has added support for multi-document ACID (Atomicity, Consistency, Isolation, Durability). Transactions in version 4.0 extended that support for distributed multi-document ACID transactions in version 4.2. MongoDB's document model allows related data to be stored together in a single document. Mongodb automatically aborts a transaction after 60 seconds. MongoDB uses multi-granularity locking that allows operations to lock at the global, database or collection level, and allows for individual storage engines to implement their own concurrency control below the collection level, which we will be utilizing for any lock based concurrency issues.
23. Competitive analysis:
24. EveryDollar.com

| Features | EveryDollar | Ours (TYM) |
| --- | --- | --- |
| Summary of the transactions | Focuses on planned money vs spent money. | Focuses on income vs expenses and focuses on savings. |
| Insights | Default graph is a pie chart with too many options to choose for viewing. | Default is a area chart that focuses on income vs spendings in multiple categories, with option to view in other graphical forms. |
| Ease of use | There is a learning curve with too many options but not many functions. | Just the basic functionalities that make sense. |

1. Mint.com

| Features | Mint | Ours (TYM) |
| --- | --- | --- |
| Summary of the transactions | Gives a overview of the money in all accounts but not total for the user | Give an total of all the saving resulting from income - expenses |
| Insights | Too many categories with no option to delete or sort the default ones. | Only essential categories with full edit functions. |
| Budgeting | Does not help to stick to a budget after setting it | After setting the budget, each expense record will remind of the budget. |

1. High-level system requirements

Below is a list of the technologies used in Team's software stack:

Server Host: Google Compute Engine 2vCPU 2 GB RAM

Operating System: Debian 10

Database: Mongodb 4.9.1

Web Server: Caddy 2.5.2

Server-Side Language: JS

Web Application Framework : Express

Additional Technologies:

IDE: VS code

Supported by all major modern browsers.

1. Team

* Robert Sato - Team Lead
* Siham Argaw - Scrum Master
* Hemanta Thapa - Front-end Lead
* Neel Manthani - Back-end Lead
* Diego Islas - Github Master

1. Checklist

​​· Team found a time slot to meet outside of the class

· Scrum Master shares meeting minutes with everyone after each meeting. **DONE**

· Github master chosen **DONE**

· Everyone sets up their local development environment from the team’s git repo. **DONE**

· Team decided and agreed together on using the listed SW tools and deployment server **DONE**

· Team ready and able to use the chosen back/front-end frameworks. **DONE**

o For each technology (front/back-end/DB/cloud) , the team decides who will lead the study of each technology and what will be output of the (feasibility) study within one month. **DONE**

· Team lead ensured that all team members read the final M1 and agree/understand it before submission. **DONE**