Project Overview

Description:
Create a spend tracker dashboard. Pulling data from previous transactions/statements. Have the tracker give historical stats, future predicted trends and what suggestions it can give for better spending in the future. Consider courses of income to help with predictions. Algorithm creation to help with future trends and considerations. Connect to other customer accounts. Connect to subscriptions services. Connect to any other place you are spending (Loans, PayPal/Venmo/Zelle), Categorize expenses. Am I using the right rewards compared to my spending and what reward programs are out there. Letting customers know which card to use at which store.

Start with real - world application - Income from PT job, assess where they are spending money on entertainment. Review cost of student loans based on potential salary.

Final Deliverable: Automated spend analysis from precious transitions.Dashboard/view with options/filters. Month by month comparison.

Logic/COns analysis of options

Gauge workload, possible homework during lab sessions

6-7 weeks of content, then build the last 1-2 weeks (mockup tool)

1st week focuses on introductions, project scope, etc

Questions (Free feel to add on to this part)

- What are the languages that we are going to use in order to make the dashboard?
- In order to create an algorithm, what would be required? Would we be using existing algorithms or make something from scratch?
- Would the use of be allowed?
- Are there any restrictions that are set in the project? Are we not allowed to do certain things or is it ok as long as we fulfill the project guidelines?

5 different departments in iit org

- Application/software development
    - Makes the application that the customers use
- Infrastructure
    - Build and maintain servers etc.
- Architecture
    - Pattern, system design,
- Security
    - Makes sure everything is secured
- Technology capability and innovation
    - Makes sure existing can upscale

Agile team

- Product owner
    - Business owner/technology group
    - Future of business
- Scrum master
    - Keeping team on track
- developer/engineers
    - run and test code
- tester/quality engineer
    - Unit testing/system testing
- Ux designer
    - Front end design

Our Assignment for today (9/26):

Project deliverables refer to all the outputs that are submitted within the scope of a project.

The point of the project is to display a chart of the categories where you spend your money and how much of your money goes to that category. The point of the tracker is to allow you to see where your money goes and give tips on how to better save your money. It will help you spend your money more wisely and where you should and shouldn't spend your money. Providing income will better determine if you are spending too much or if you are spending too little, because it will tie back to whether that category is a necessity or just a want. The pie chart will be able to set budgets on each category and will help guide you to what you should be buying and how much you should be saving based on your income.

Answer questions like:

- Am I using the right rewards compared to my spending?
  - Have a variable that calculates the total amount of rewards you are supposed to receive and compare them with what rewards you actually received

- What reward programs are out there?
  - With Every category there must be rewards linked to it that show up as soon as the person clicks on that particular section on the chart a set of rewards pop out on the screen and offer ways to reduce the amount.

- Am I spending too much on a particular category?
  - Have an available pie chart indicating what categories you spend most on (pie chart depicts % and total $ spent per category). The pie chart can declare what categories are necessities and what are wants.

- What are some suggestions for savings?

- Differentiating between your needs and your wants can help you avoid spending on unnecessary purchases.

Outline of Week 2

In the agile process, before we start building/creating code, we want to make sure we know what we need to do and why. The intent of answering the following questions will help us understand what we need to create.

What is an Initiative? An initiative is a request for a new project with the following summary detail. Once an initiative is created, smaller units of work can be created, called Features or Epics. Let's create an initiative for each group and then start thinking about features.

Create Initiative:

1. Description. 1-2 paragraph description of what we want to create and why.

2. Requirements. Before a project begins. We want to have clarity on what functionality is needed. Create bullet points of all requirements. This is not the how, but what we want to have at the end of the project

a. Example. Ability for user to know how much money they spent in the past 12 months.
3. Success criteria. At the end of the project, what does success look like. What needs to be completed to consider the project a success. This is important to establish up front to keep the project inline prevent scope creep). And we can look at this at the end to make sure we checked all of the boxes. Create bullet points for success criteria.

a. Example, Code does not have any bugs, all test cases are completed.

Create Feature/Epic I

4. Build. What are we planning to build and how? Are we creating an actual program. Or going to mimic code in an excel/word/power point document. We should understand at least in general what is needed. Create an outline of what we need to successfully build the project, use bullet points for now.

5. Test. Write test cases of what we need to check to make sure the build is working as expected and does not have any bugs. Start with bullet points now, we can create details of each test at a later session.

Summary of Week 2 (Link 🔗)

- Discussed the description of the project.
- We went over some of the requirements needed for the project and how the project should be displayed and how it should look like. Some of the requirements included how the program should be programmed, like how it should be made. Things that need to be included are
    - User inputted income to help track and make a budget.
    - Identify the names of each category.
    - Label the categories based on whether the category is a need or a want to help budget your money more wisely.
    - Have the project tell you what categories you need to spend more or less money on.
    - A display of pie slice sizes to help indicate where most of your money goes and the size also indicates the amount of money each category needs to be receiving.
    - There must only be one display of that user's pie open at a time for safety and security measures.

- Divided the process of making the project into five main parts and assigning every team member one of them.
- Identified the things which are out of scope for the project (the things we do not need to consider)
- Not included in the project: user credentials (login username / password)
- How do we determine the success level of the program?
    - We can determine the success level of the project by examining how the pie looks and the data outputted. For example, if the pie chart accurately takes in the income inputted and distributes that money across the categories at different amounts depending on the category importance and having some saving money, then we know the data inputted and output is correct. This is only one example.

Summary of Week 3

- We discussed what APIs (application programming interface) are and how they are used.
- We created a drive consisting of all our resources that is shared amongst team members.
- We were provided a document with a list of resources we can use:
    - Technology discover (website)
    - A Business User's Guide to APIs (video)
    - C4 Diagramming Model (website)
    - Wireframes (website)
    - draw.io (website)
- We learnt how to make a wireframe and allow all members of our project access to it.
    - It allows all members of the project to add any sort of data to the wireframe in live time. This allows for better collaboration between team members and allows everyone to have a decision on what is needed and what isn't. Allowing multiple teammates to access the wireframe at the same time speeds up any process of the project.
    - The wireframe allows us to create and visualize what the outcome of the project will be. It builds a foundation of what we will need to do on the back end of the project.
    - The wireframe gives a base of what the programming part of the project will be. Almost a pseudocode for the project, but rather it is a display instead of english words.

Summary of Week 4

- We learnt how to make our project more interactive by “adding API specifications.” For example, clicking on a certain category will only give the transactions and the graph of that category specifically, so we can see how much is spent in that particular category along with how often one shop/spends there. Viewing transactions will give you the merchant’s name, and the amount deposited or withdrawn from the account.
- Making the wireframe look more like the outcome and having it as detailed as possible will help more and more in the backend of the project because it gives us the base of everything we will need to do.
- We’ve started to learn how to use a website called ‘Spring Initializer’. It will be the platform that we will be using for the coding portion of the project. It will help with the packages and other things we will need to make the project. Will be able to push the coding we’ve done into GitHub so that other members of the group are able to access it.
- GitHub can be where we access the coding so all other laptops will be able to run the code and make changes where needed.
- Lastly, the spending dashboard needed to be fixed and revamped to be more appealing and have greater functionality.
- We have made the wireframe look more pleasing to the eye instead of it being a boring display. We want the users to be more pleased with the look so the app doesn’t strike the user as being a boring savings app with no eye candy. The purpose is to have the users want the app because of the way the app works, looks, and feels. Looks isn’t a big thing, but it makes the user understand the time and effort that was put into the app.

PROJECT RUBRIC

What is the problem statement?

Answer: The problem is that users are lacking good money management through their usual banking apps. We want to create a dashboard application that will better help users with money management and help them keep track of their spendings with easy to read data and visuals.

Explain the process your team went through to develop your solution.

Answer: The process we went to develop the solution was to first create a wireframe that displays what the outcome of the project will be. It allows us to understand what we need to create and what programming we need to do to solve the problem stated. Being able to have that wireframe gives us details of what we will need to do and what programming we will need to do on the backend of the proposed project.

What is your team's proposed solution?

Answer: The solution is to create a spendings tracking dashboard that will be user friendly and very interactive to help the user understand where his/her money is going. The dashboard will allow the users to understand where to put more/less money and give live transactions for each category. It helps the user budget his/her money according to the given imputed income.