

0.a Come up with a team name for your group.

CSEnjoyers

0.b Please list the names and PIDs of the team members who are present today (or knowingly absent)

Max Stroh

Dheer Prajapati

Khoi Lam

Akash Dubey (akashd)

0.c Provide your preliminary project idea (or set of ideas). This is not a commitment to a project.

Budget Buddy: is a Finance app that keeps track of all your expenses that you have made in the past week, it also has functionality to allow users to limit how much they spend in a day and a week.

1. Provide an example of five hypothetical non-functional requirements for this system. Be sure to include the specific type of requirement discussed in class, with each requirement coming from a unique category.

- 1. Usability: Must include text that is easy to read/see.**
- 2. Supportability: Should be able to be updated with ease and be able to change for the better based on user feedback.**
- 3. Reliability: Should always provide accurate data and save important information from the user.**
- 4. Performance: Should run smooth on most devices for all users.**
- 5. Implementation/Constraints: Must use python and/or Swift(for IOS).**

2. Provide an example of five hypothetical functional requirements for this system.

1.) A login system

2.) Secure way to access users bank accounts

3.) Users will be able to use the apps budget feature

4.) Users will be able to get help if they come across a issue

5.) Users will be able to reset their password if they get locked out of their account.

3. Think of a specific task required to complete each of the functional requirements and non-functional requirements mentioned above (10 total). Estimate the amount of effort needed to complete this task using function points (i.e., using the values [here](#)). Briefly explain your answer.

A specific task that is required to complete each of the following functional requirements and non-functional requirements would be the following:

Non-functional requirements

1.) make a nav bar so users can navigate to the setting and alter text settings. This won't be extremely challenging since there are a lot of helper tools for this.

2.) Make the software in a way that it can be altered with ease, therefore we would use an agile scrum methodology. This will be challenging since user have a lot of freedom

to manipulate the pages in the software, as a result, coding this will require a lot of bug fixing and thinking outside of the box

3.) Make sure we use a database that is familiar to all members of the team. This won't be challenging, the hard part is the gathering the database itself.

4.) Program the application to be viewable for all form factors and make sure the front-end code is written to be friendly for smaller and bigger screens. This will be not hard nor easy since a workable application is not hard but a good one is.

5.) We will have to learn python and Swift if we are not already familiar. This will be not as hard since if you already know a language, you know the foundation for others.

Functional Requirements

6.) Make a web API that allows the user to login with the use of a database to store password and username credentials. This won't be as hard as the back-end, however, it still requires good knowledge of web-design and cryptography.

7.) Will have to talk to various banks to forge a deal with their data. This will won't be hard on the technical part but it's really hard to get a good deal.

8.) Will need to program a back end and a front end for the budget system, make it so it is user friendly on the front end and robust enough to handle random edge cases users might try. This is the hardest part since it's the skeleton structure of the whole software.

9.) Implement a help page. This will be manageable since it's not code heavy and we can just put a phone number in.

10.) Implement a reset password page and system where the user gets an email with a reset password link. This will be not hard nor easy, however, it's still manageable since it's nothing out of the extraordinary.

4. Write three user stories from the perspective of at least two different actors. Provide the acceptance criteria for these stories.

- As a housewife, I want to use this Finance app so that I could organize groceries bills better. Acceptance criteria will be showing the spending daily, display graph corresponding to the trend of spending.

- As a business, I want to use this Finance app so that I can keep track of my money and make sure no one scams me. Acceptance criteria will show the spending daily, notify the customer when they are charged for something.

- As a teenager, I want to use this Finance app so that I can learn how to manage money better. Acceptance criteria will display a graph corresponding to spending, show future spending based on the current trend.

5. Provide two examples of risk that could potentially impact this project. Explain how you would mitigate these risks if you were implementing your project as a software system.

One risk that could potentially impact this project would be having people's personal information/finances being leaked or easily exposed. We need to ensure that our app is secure, as well as people's information when tracking their finances.

Another risk that could potentially impact this project could be the scalability of this app. If there are a lot of people that sign up for the app and use it, we need to ensure that the app can continue to keep up with the traffic and perform accordingly. We can do this by optimizing our databases that we make, which can allow for more data to be held.

6. Describe which process your team would use for requirements elicitation from clients or customers, and explain why.

For requirements elicitation from clients or customers our team could use a combination of processes. Given the initial ideas and potential risks identified, conducting interviews and surveys with potential users would provide valuable insights into what features they value most in a finance app, their concerns about security, and how they currently manage their finances. This would help in refining both functional and non-functional requirements by gathering detailed user expectations/preferences about using such an app.

Our team can also go through stages of prototyping. Creating prototypes of Budget Buddy and presenting them to a group of potential users can be an effective way to elicit feedback on the app's usability, design, and features. This allows us to test usability by allowing users to interact with the app's interface and providing feedback on ease of use. It also offers insights into how well the app meets users' needs, which can be crucial for refining functional requirements.

This multi-faceted approach ensures that your team gathers a broad and deep understanding of user needs, preferences, and behaviors. It is particularly suited to the development of Budget Buddy because it addresses the complexity of managing

personal finances and the critical importance of security, reliability, and usability in financial apps