Keith Kaczor

COMPSCI 482

Project Proposal

25 April 2016

**Executive Summary**

*Please provide a summary of the information detailed throughout this project proposal.*

My application proposal is a Budgeting and Bill Reminder app. It will have various budgeting features, and will be based around a user-created account.

**Statement of Need**

*Discuss the needs or opportunities to be addressed by this project. In your discussion, describe and position these needs or opportunities within the context of your organizational environment. Internal factors, as well as external ones, should be considered.*

Budgeting web applications like Mint have a lot of great features that can be duplicated, but one thing Mint does not have is a way to know how much money you will have in you account after bills are paid. My app will generate the next number of bills you have for a specified period of time (probably a month), and then subtract the total of those bills from your account’s funds.

**Project Goals**

*Identify the goals of the project. Discuss what is to be achieved and the expected/desired outcomes of this project.*

The goals of this project will be to give a user information to help with budgeting and bills by showing the upcoming bills and displaying the relevant account information. Users should at least be able to add or change the amount of money in their account and create bills with dates. Acceptable outcome would be something like what is in my prototype, but with more features.

**Constraints**

*Talk about constraints that could affect the development, implementation, and/or end outcomes of this project. Consider such factors as experience, knowledge, skill sets, budgetary resources, time, competitive environment, and existing investments.*

One constraint is lack of any direct access to real-life account information. Competitive services like Mint allow you to actually connect various financial accounts together and show them as one consolidated number that is updated when money is deposited or withdrawn from those respective financial accounts. My application will be limited to interaction based on direct user input because I do not have the required skills to implement the connections to a user’s financial accounts through the various APIs that likely exist.

With a large amount of time, it is possible I could learn to implement these sorts of features, but with the time constraint of this project it is not feasible. I also assume there are guidelines and regulations for any financial institution’s API, which would take even more time.

Ensuring user account security may be attempted with this app, but similarly to my SmoothieWiz app from 481, I may not have the skills to implement any sort of encryption for things like user passwords.

**Requirements**

*Given your operational context, describe what needs to be in place during the project and afterwards in order for the project to be successful. In your discussion, consider the constraints you identified above and, if necessary, define requirements to compensate for constraints.*

For this project to be successful within the constraints listed above, it will need to allow the user to directly interact with their account information and send it to the server-side datastore. Even though I have a fairly good understanding of the datastore, proper and efficient datastore queries are key.

**Evaluation and Measurement**

*Propose criteria that might be used in defining and evaluating the success of this project and its end outcomes. What measurement standards are available to determine if the project is a success and the impact of its end outcomes?  What benchmarks can be set up to determine the short-term and long-term efficacy of the project? Consider in your discussion the goal and, requirements defined above.*

Fundamentally, this project’s shore-term efficacy is reliant upon a user being able to create an account, add funds to their account, create bills, and have those bills be displayed in a way that includes future account funds after those bills have been paid.

Long-term efficacy is simply adding more information and features, like calculating of debt and managing loans potentially.

**Potential Solution**

*At a high level, describe a potential solution that would satisfy the goals and requirements defined above.*

 Storing of information must be done server-side with GAE’s ndb.models. Account funds will be stored as ndb.Integer, Bills will be stored as a ndb.Struct in the User model, and have their own model to call with the Struct. A potential problem I may have with a Bills model is querying them down to only a specific user’s bills, instead of querying all bills.

**Alternative Solutions**

*Do you know of any alternative solutions that satisfy the goals and requirements of this project (either fully or partially)? If so, have they been explored?  Discuss why this proposal has been submitted in lieu of alternative solutions.*

No alternatives known.

**Contingency Plan**

*If the proposed project is critical to your operations, please discuss what contingency plans could be put into place in the event that the project or its end outcomes are not successful.*

Most important to contingency is getting the future account balance after bills have been paid. This is the defining feature of my application in comparison to the competition, so it must be able to show this in a way that is easy to read.