

Focus Manager

Design Document

Author: Phillip Thain

Last Updated: 06/21/2019

“A design doc is the most useful tool for making sure the right work gets done.”

-Angela Zhang

Problem Statement

Working independently on large projects can be a challenge because of the long term nature of their creation, as well as their inherent complexity. The goal of this particular project is to implement a piece of software that will organize projects and stimulate productivity of independent work to ensure that long term goals are achieved in a reasonable amount of time.

Background Information

This project is a personal endeavor. I have a tendency to start work on projects, only to put them on the shelf and never return to it. Therefore, this software is an attempt to give intangible ideas enough structure to become a reality.

The ultimate intent of this application is to create a “work environment” on the computer. While plenty of “to-do list” applications exist, most are designed as general purpose tools to be used throughout the day. The intent of the Focus Manager is to be used specifically for work. It will allow users to not only write down *what* needs doing, but creates space to describe *how* to do those things, and track *how long* they take to complete. Visually witnessing progress on something normally intangible is extremely motivating. That motivation is necessary to keep moving forward when working independently.

Design Outline

The Focus Manager is an application that will be able to organize large projects that a user creates and customizes. The application will provide the following features for creators:

- Project creation & management
- Time tracking
- Data visualization of progress

Focus Manager will be implemented as a cross-platform desktop GUI application (including Windows, Macintosh, and Linux) using Javascript and the Electron Framework.

Design Issues

Functional Issues:

Non-functional Issues:

1. How should we store the information?
 1. If we have any notion of “saving”, and we want something less transparent than a FLAT file system, then it requires a database which may benefit from a server in the case that the program is extensible to multiple users and devices.
2. Why can't I focus on this question?
 1. It is hard!
 2. I don't even know what information needs to be stored
3. What information needs to be stored?
 1. Projects, Goals, and Tasks
 - 2.
4. Should “goals” and “tasks” be abstracted to one object called “items” that can be labeled as “goal”, “task”, “problem”, etc.?
5. Should the project interact with a server or database (AWS, or local storage) for managing project information?
 - Pros: - Using something like AWS makes cross-device interaction easy
 - Cons: - The work to build a database like that may be overkill for what can be solved with a few configuration/storage files

Design Details

Class Diagrams: DONE

Sequence Diagrams: → in progress

Activity Diagrams: → in progress

Sample GUI Elements: DONE