Chrome Tasker

Project Proposal by Alex Bochman

What Is Chrome Tasker, and What Is the Motivation Behind It

The idea for Chrome Tasker is that it will be a Google Chrome extension that acts as a tool to track, manage, and provide additional information on user created tasks. The fact that it's a Google Chrome extension means that **anyone** with Chrome installed on their computer can quickly install the extension from the Chrome Web Store[1] with little to no setup or system requirements. Additionally, the idea behind Chrome Tasker makes the extension versatile because it can apply to common everyday tasks, complicated routines performed by developers or engineers, and everything in between.

Chrome Tasker will be designed to gain insights into workflows by tracking specific tasks and providing users an estimation of how many times the task could have been completed based on user input data. The idea for this project stems from my IT internship at Parker Hannifin, a motion and control technologies corporation. There, my boss was working on something similar that helped supervisors track the workflow of machinists and automated robot arms. After expectations weren't met, the data gathered was then used to improve numerous aspects. The fact that the expectations weren't met helped this large company identify faulty equipment, employee schedule conflicts, and other anomalies that were unaccounted for. Once the issues were addressed, productivity increased. This project aims to achieve the same results with a much broader audience as a Google Chrome extension.

There are many ToDo lists and task trackers all over the web, including Google's Web Store, but I haven't found anything quite like this proposal. While everyone may not use every feature that I'm proposing, it can be used in a variety of ways by individuals and groups alike. The best part is that it'll be free and accessible for everyone and anywhere there's a desktop. The features below are unlike any I've found in any other task tracker.

Key Features

Google Chrome extensions are developed using Chrome APIs[2] in addition to web development technologies: HTML, CSS, and JavaScript. It'll be easy to get started developing this project with web technologies; all you'll have to do is grab your favorite text editor, get started with Google's API documentation, create a few files, and then publish the extension to the web store. Let it be noted that it does cost a one-time fee of $5 in order to publish extensions on Google's Web Store (that's only a little more than a coffee or two)! The key features that will be essential to Chrome Tasker's operation will be listed below:

---

[1] https://chrome.google.com/webstore/category/extensions?hl=en
[2] https://developer.chrome.com/docs/extensions/mv3/getstarted/

- **Tasks**
    - User can create, select, edit, delete, and mark tasks complete.
    - Active and completed tasks will appear in a sidebar on the left-hand side of the page.
    - A task that is selected/highlighted designates it as a current task, and will open up the task analysis options and information for that task.
- **Task Analysis**
    - A selected task will open the main panel to the right of the tasks sidebar.
    - Essential / key options for the given task include:
        1. A timer that can be started, paused, and reset. Implementation flexibility regarding whether the timer can be set, counts up, and/or counts down.
        2. Roughly how long does it take for the task to be completed? User input box provided.
        3. How many repetitions will be performed? User input box provided, as well as an option for an indefinite amount.
        4. How many users are performing the task?
        5. How long will each user work on the task? Must take breaks into account.
    - Bullets 2-5 above will be essential to a final calculation that will be displayed that lets the user(s) know how many repetitions should have or could have been completed.

Optional and More Complicated Features

These features will be optional and/or more complicated, and should only be tackled after the key features. Additionally, I've provided some of my initial thoughts at the time of writing this document.

- Implement a function that accepts user input emails. If enabled, this feature will send out an email if a condition is met, such as the user or users completing their tasks.
- Implement a historical record of tasks, or a historical record of a single task. Leaves the possibility for graphs to be implemented.
    - Another option would have to be implemented where users input the amount of task repetitions that have been completed after some amount of time. This number would have to be stored in order to have a historical record/graph of the task.
- How will saving files, tasks, and user input work? Where will the data be saved?

Success and Risks

Teams working on this project will know they've succeeded when they've implemented most or all of the key features. The two major ideas behind Chrome Tasker are to **add tasks to some sort of list** and to come up with ways to **calculate how many repetitions of that task could have been completed within some amount of time**. The single most serious risk that I see in developing this project on schedule will be **learning Chrome's API** which is to be used in JavaScript. Aside from the risk, groups working on this project will need: HTML, CSS, and JavaScript knowledge; and $5 to be able to publish to Google's web store.

Many things in the mock-up are subject to change including but not limited to: the theme, button and information layout, and the available features. Mock-up created using Axure RP 9[3].

Hours spent on this document (not including lots of brainstorming): 4.5 hours

Hours spent on creating my presentation:

---

[3] https://www.axure.com/