

Potato Productivity

For when you feel like a couch potato.

Professional Piscators: Alan Chen, William Xiang, Vincent Liok, Julius Freyra

Pitch

Have a lot of stuff to do? Don't feel like doing it? Here is Potato Productivity, tailored to suit all your needs, wants, dreams, feelings, and more! With a Pomodoro timer (and mandatory breaks. yes, that's right! MANDATORY BREAKS!), you can plow through all those tedious chores in no time! By combining hardcore balls-to-the-walls no mercy work sessions with a musical playlist of your choosing, you can now plow through those dirty differential equations with ease! Losing steam? No worries! The Potato Productivity forces you to take breaks in between work sessions to help declutter thoughts and help you get ready for the next problem pounding session!

About

Potato Productivity is a productivity tool that utilizes a user database and the Spotify API to help the user stay productive. The main feature of the site is the Pomodoro timer, which times the user while they work and notifies them when they should take a quick break to keep their mind fresh and prevent burning out. There will be a dynamic task list on the side that will allow the user to add tasks as they come up and delete them as they complete them. The site will also implement the Spotify API and generate either a random work playlist or a playlist based on the user's preferences.

Roles

Alan: Frontend (HTML [Foundation], Jinja, Javascript)

William: Middleware/Frontend (Python Flask app, Javascript)

Vincent: Backend (SQL, Python backend)

Julius: Backend (API, Python backend), Project Manager

Components

1. Pomodoro Timer
 - A timer that sets a certain amount of time for a user to work and a certain time for a user to rest
2. To-do list
 - A dynamic list that allows the user to add, remove, and organize their tasks
3. Playlists
 - Spotify driven playlists that offer playlists to help the user relax during a break or work during the delegated working period
4. Flask App (app.py)

Middleware Python file that takes parses and directs information to where it needs to go, from the client end to the backend and vice versa

5. User Database

Database containing the user's login information

6. Database Manager (auth.py)

Parses information from the user database to send to the frontend; updates the database when the user changes or adds new information

7. Spotify Manager (spotify.py)

Calls results from the Spotify API and creates a playlist for the user to listen to

8. Timer (timer.js)

Controls the dynamic clock

9. To-do List (app.js)

Controls dynamic action of the to-do list

10. HTML pages

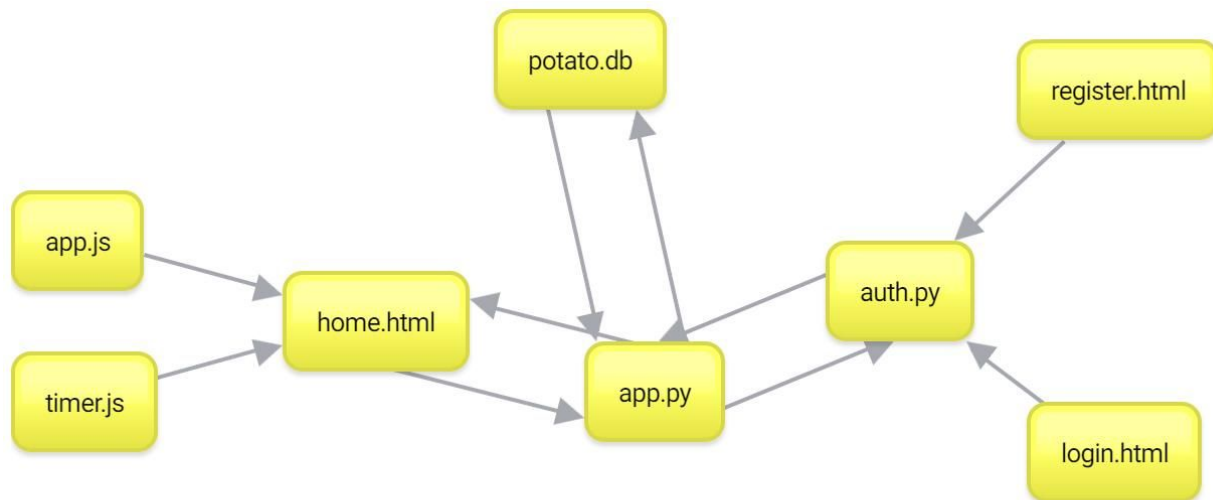
home.html:

Directs the user to login/register or to start using a standard pomodoro timer, dynamic to-do list, and music search.

login/register.html:

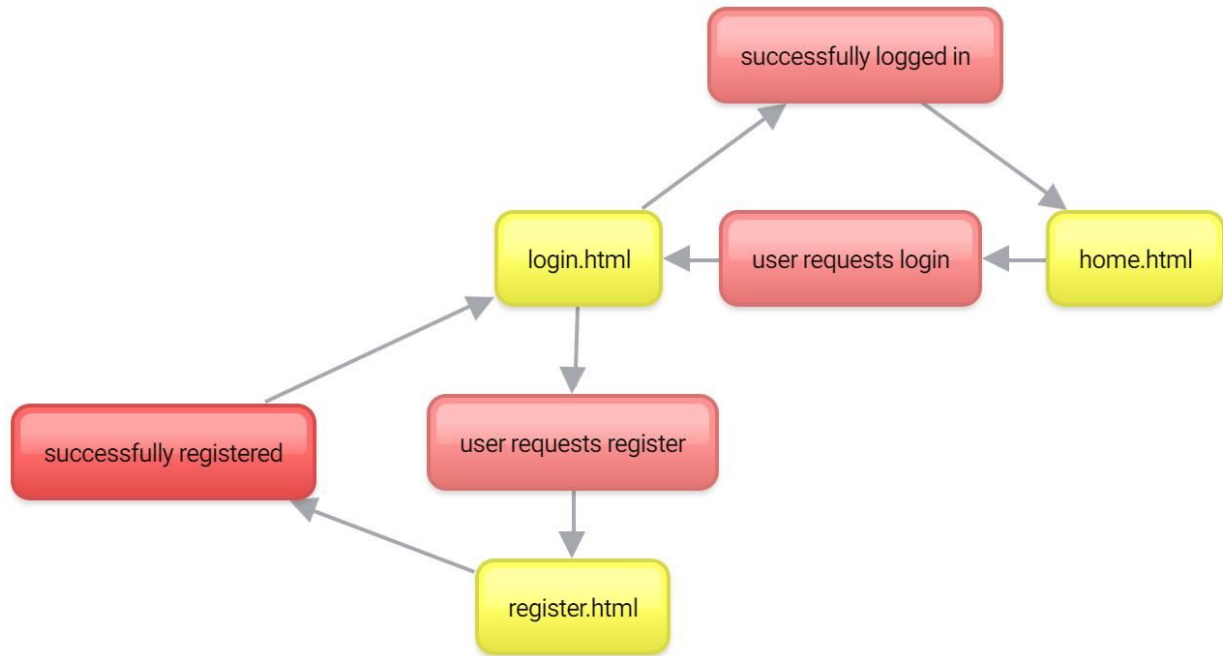
Basic login/register functionality

Component Map



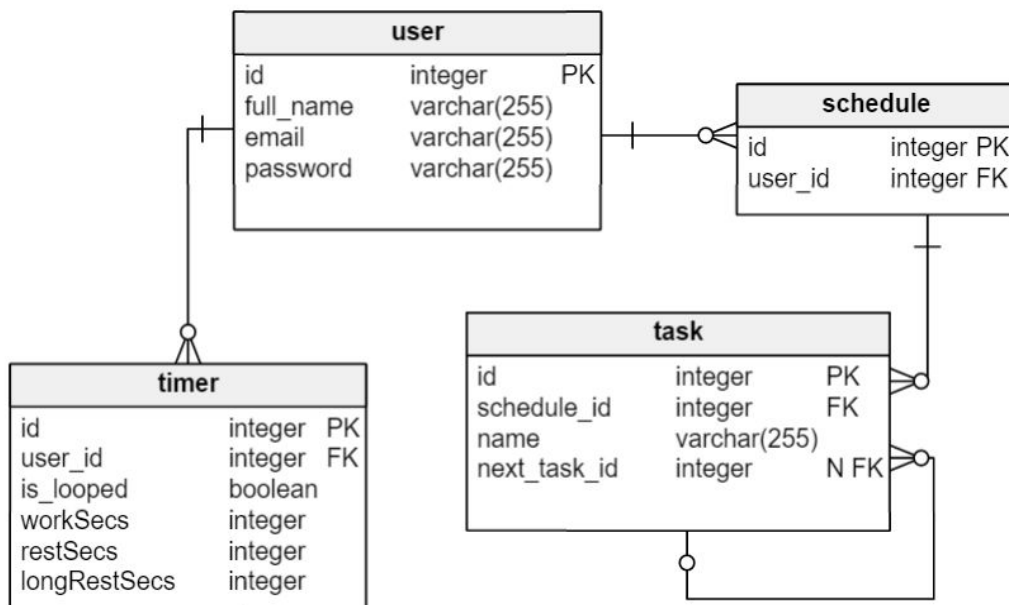
created with www.bubbl.us

Site Map



created with www.bubbl.us

Database Schema



Page Mockup

Potato Productivity

24:59

Pause

No pausing allowed. Maximum Effort!X

To-do: Save

Task name:

Spotify

