

Git Güd

Project Description:

This website allows someone to take advantage of the new year to git güd — that is, to expand one's vocabulary and improve their (touch) typing speed. If they've mastered all that, they can git güder by checking out random activities we have on our site and getting more involved in the world around them.

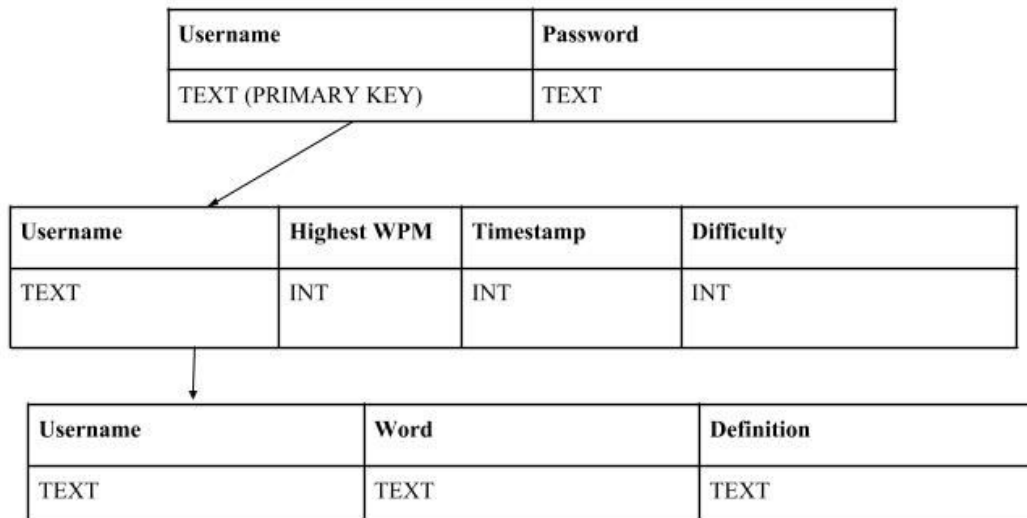
APIs Used:

- Bored API
- Quotes API
- Words APIs
 - Datamuse API
 - Oxford API

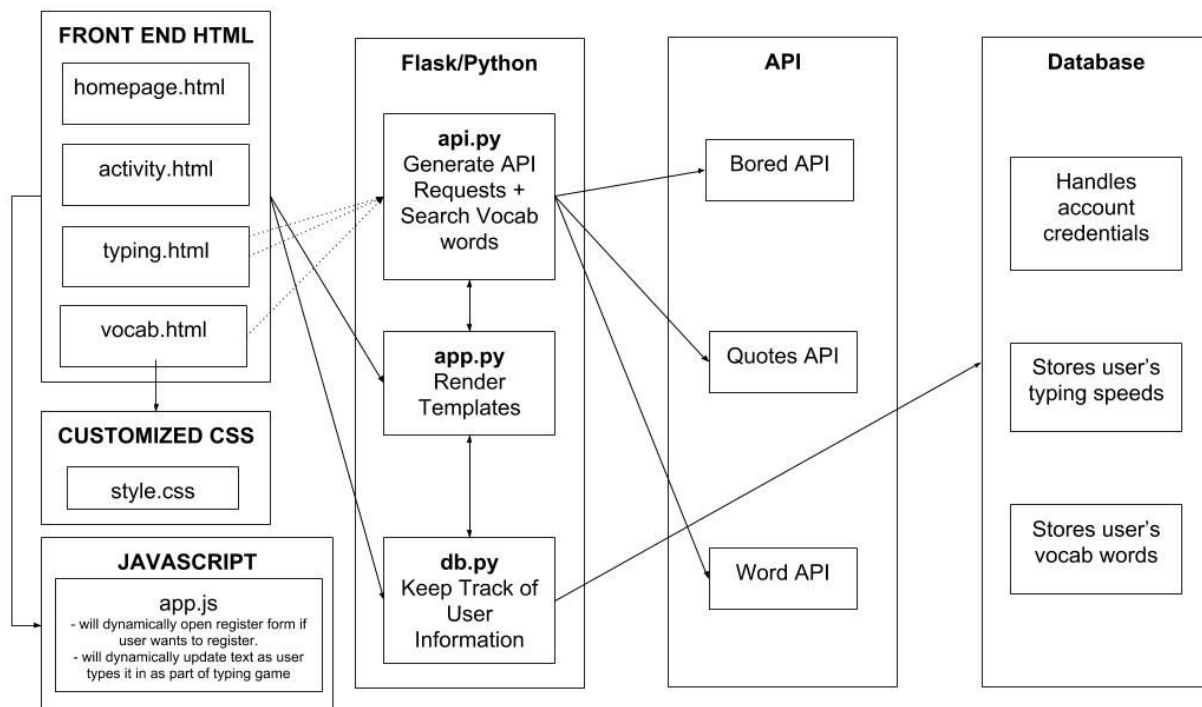
Features:

- Front End Framework
 - Foundation
 - Explanation: We all had experience with Foundation from our previous projects, and decided to stick with what we were experienced with.
- Random activities
 - Using the **Bored API**, random activities will be generated for the user to do.
 - Possibly a To Do list.
- Words
 - Initial UI: The list of words the user selected already and a search bar
 - A user searches for a word. Different words will be returned using the **Datamuse API**. Each word will have an accompanying definition using **Oxford API**.
 - Users can select words that will then be displayed on their Dashboard. He/she can return to those words by going to their Dashboard at any time.
- Typing
 - Initial UI: The fastest WPM for each difficulty and a "Select Level of Difficulty"
 - The user will be given a paragraph/quote to type as fast as they can.
 - At the end of the stage, the user will be shown their WPM for that level.
 - Database will be updated if the wpm is higher than previous record.
 - There will be different levels of progression (such as adding numbers, hard to reach characters, symbols, etc.)
 - The **Quotes API** will be used for some of the levels.

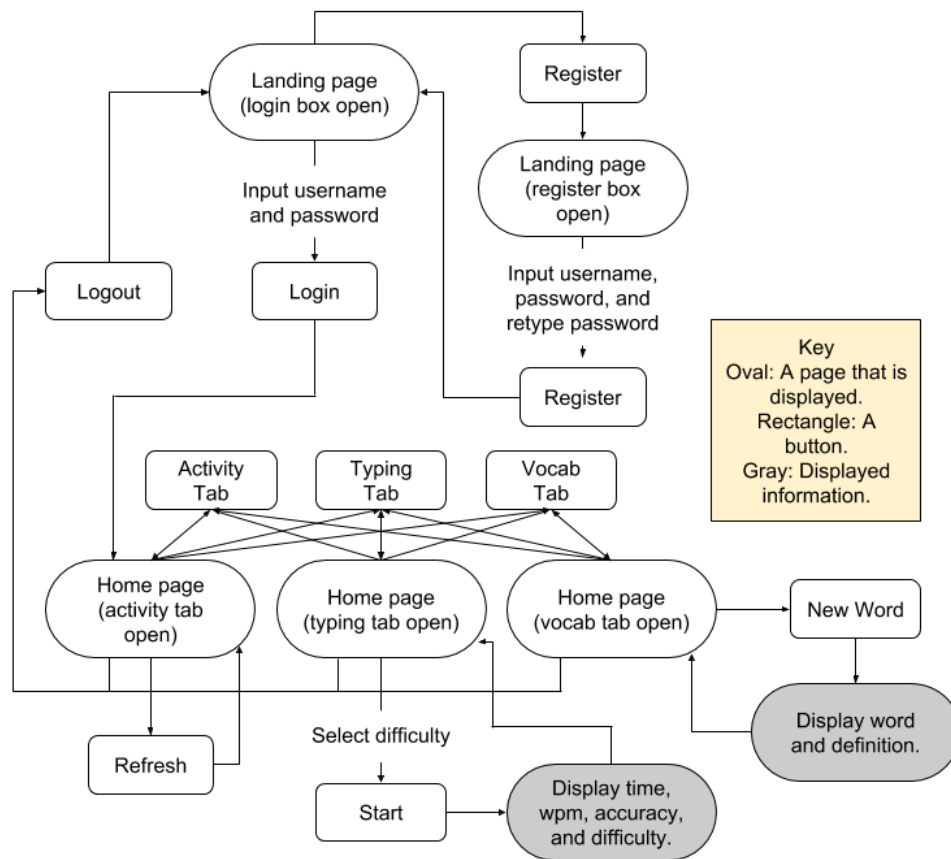
Database Schema:



Component Map:



Note: Begin on Landing Page (login box open)



Task Delegations:

- RAUNAK:
 - Words
 - Flask work
 - Frontend Design (% TBD)
- HUI MIN:
 - Random activities
 - Database work
 - Frontend Design (% TBD)
- ANTON:
 - Typing
 - Frontend Design (% TBD)

Roadmap:

2019-01-07

- Database skeleton set up (H)
 - DB creation (and checking for the existence of the DB)
 - All current DB calls should be functional
- Login system set up (H)
 - Fully functional, can use login system from last project
- API skeleton set up (R)
 - Using generalized API call function from the last project
- Flask skeleton set up (R)
- Foundation skeleton set up (A)
 - base.html, etc.
- All necessary Python files created (H+A+R)

2019-01-08

- Begin implementing DB calls according to schema (H)
- Begin working on Words (R)
- Implement necessary routes up to this point (R)
- Begin working on Typing (A)

2019-01-10

- Finish Words (R)
- Implement necessary routes up to this point (R)
- Continue working on Typing (A)
- Implement all necessary DB calls up to this point (H)
- Depending on who finishes, work on frontend HTML/CSS (H+A+R)

2019-01-11

- Hard deadline for Typing (A)
- Implement necessary routes up to this point (R)
- Implement all necessary DB calls up to this point (H)
- Depending on who finishes, work on frontend HTML/CSS (H+A+R)

2019-01-13

- Stress testing complete (H+A+R)
- Acquire >8 hours of sleep

2019-01-15

- Final deadline + demos

Note: All deadlines are tentative