Team PythonScript
Tianrun Liu, Bo Hui Lu, Brian Lee, Tim Marder
SoftDev1 Pd6
2019-01-08

Final Project - "Mykolyk's Ultimate Dictionary" derived from MUD, the inspiration for the game

Plot: A crazy kid found a magical book in the QAF after failing horribly in a spelling bee competition. Out of anger, he used the power of the magical book to convert all words in a dictionary into nameless monsters so that no one can make fun of him for spelling something "wrong." The hero(user), was the winner of the competition that the evil kid lost in, and wanted to restore peace to the world by defeating the evil wizard. Along the way, the hero restores words into a new dictionary("The Mykolyk Dictionary") by beating a puzzle every time he encounters a monster. Our hero's IQ grows every x Number of words that he restores, which makes it easier for him to defeat the nameless monsters.

COMPONENTS:
1. A dungeon crawling game displayed on a website
   a. Similar to Discworld MUD http://discworld.starturtle.net/lpc/ but with plot mentioned above
      i. Try [telnet discworld.starturtle.net 23] in the terminal to get an idea
   b. TUI - text-based user interface
      i. Use javascript
      ii. User types command as he goes (type "west" for going to west, wasd, etc)
   c. Map Generation
      i. Random Dungeon map
         https://medium.freecodecamp.org/how-to-make-your-own-procedural-dungeon-map-generator-using-the-random-walk-algorithm-e0085c8aa9a
2. Details of the game
   a. The user must move around a map and collect words. When the user finds a word, they get a multiple choice question about the definition of the word and must solve it correctly to collect the word.
3. Multiplayer features
   a. Group Chat (https://scaledrone.github.io/javascript-chat-room-tutorial/)
   b. Profile page
   c. Forum for users to talk about the game
   d. ? Q/A page (good for notifying players about confusing parts or bugs) **May fit in our forum idea
4. Database
   a. Store user login info

b. Stores the game state
5. Web API
   a. IPAPI - manage users, autologin
   b. Dictionary API

DATABASE SCHEME:
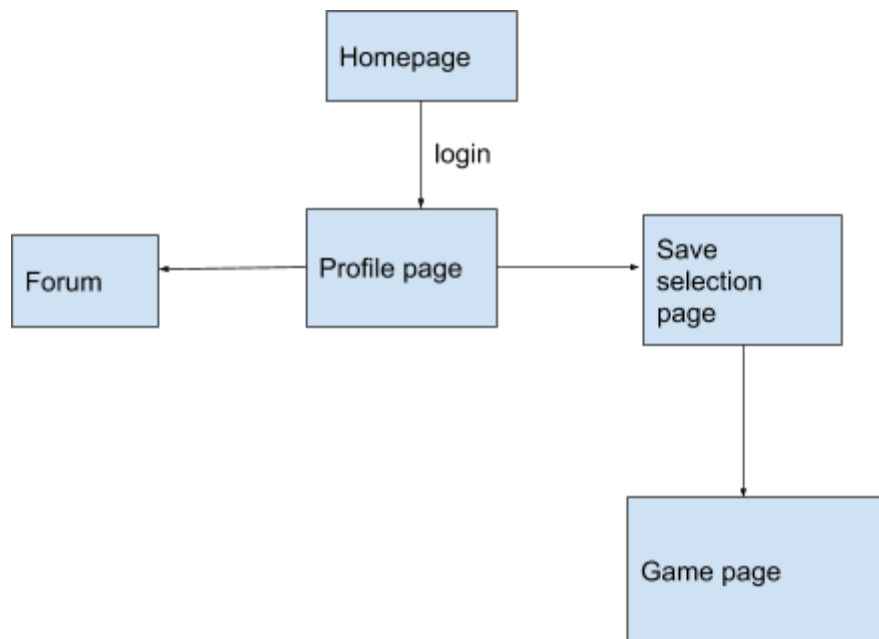
USERS

user_id(INTEGER PRIMARY_KEY)	Username (TEXT)	Password (TEXT hash)

WORDS

user_id(INTEGER PRIMARY_KEY)	word(TEXT)

SITE MAP:

TASKS:
Tim: forums/profile/maybe Q and A
Bo: dungeon generation, database, API
Tianrun: core functionality of the game(TUI, movement, etc)
Brian: Chat room, CSS, API

Component Map:

App.py
- Begins program
- Communicates with DB

IPAPI
- checks IP for logging in

login.html
- login/account creation

Users Database
- compared to inputted info from login.html

forum.html
- can chat w/ others

Homepage
- can choose to go to forum, game, or profile

Game
- starts playing from saved selection

Profile.html
- check progress