

## Sonic Heroes

---

**Fortune Soleil** - Assistant Regional Project Manager

**Julian Atkin** - Lead Socketeer + JavaScript Jabroni

**Jordan Chan** - Python- & Flask-Oriented Youth, Bootstrapping Young Lad

**Lawrence Liu** - CEO of Database Maintenance and Interaction (Ltd.)

---

### Pictionary

This will be a multiplayer online game in which one player is given a random word, and has 60 seconds to draw that word while all other players attempt to guess what word it is based on the drawing (think pictionary). The player who creates a room is able to specify the number of points to win, in order to customize game length. Each game room can accommodate 2-5 players. Words will be selected randomly from a hard-coded table in a database. Additionally, players can view their number of wins and their current skill level (determined using ratio of wins to total games).

An account is not necessary to play the game, only to have a reserved username and record number of wins. Random guest names will be assigned to those without accounts.

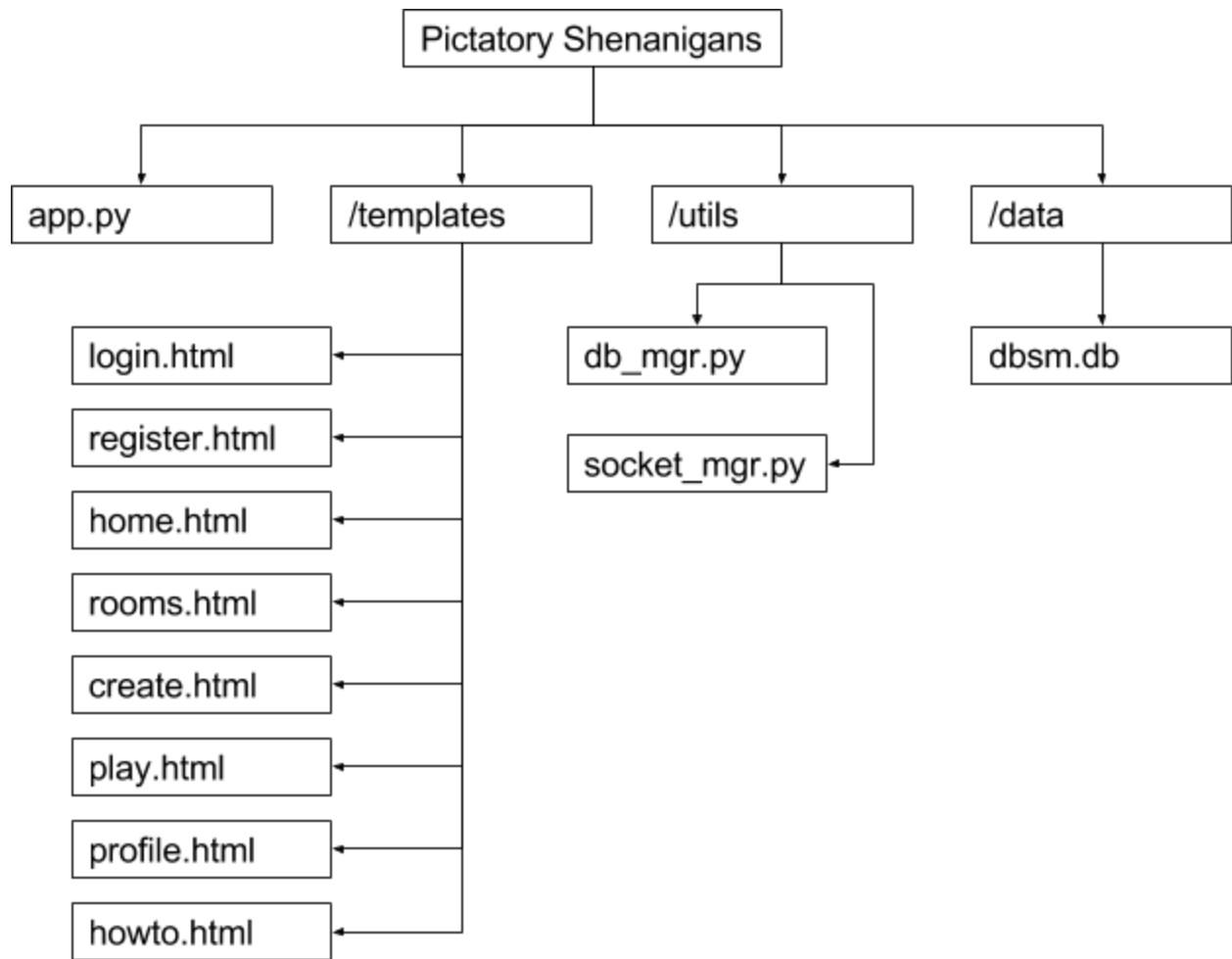
The formatting will be done in Bootstrap.

The program will use the flask-socketio and eventlet libraries.

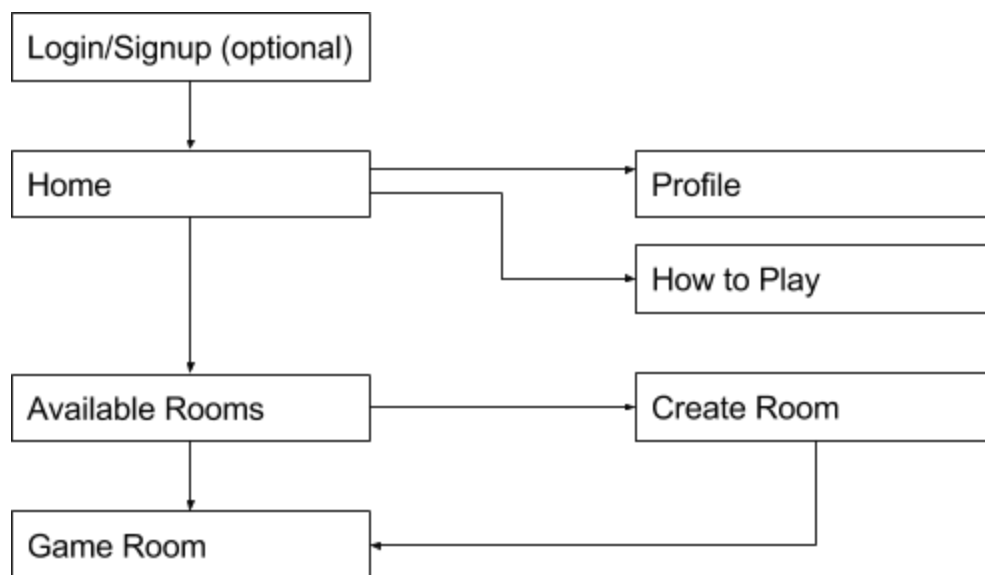
**Game Specifics:** The game will not be able to start until at least two people are in the room, and at most five can join a single room. Guesses are input in a chat box, but the guesses of other players are censored. The first player to guess correctly earns two points, while subsequent correct answers earn one point. After time runs out, the chat box is visible, as well as the correct answer and who earned points that round; if nobody got the correct answer, no points will be awarded that round. Each round, the drawer rotates; once everyone has gone, the order is shuffled and the game continues.

### Components:

- app.py - Routes: /home, /login, /register, /rooms, /create, /room/[roomid], /me, /howto
- dbsm.db - user data, potential words, open rooms + members
- html files - home, login, register, rooms, room creation, play
- Additional utilities as needed - db\_mgr.py, socket\_mgr.py



Site Map:



Database Schema:

Users Table

TEXT username	TEXT password	INT wins	INT gamesPlayed
---------------	---------------	----------	-----------------

Words Table

TEXT noun
-----------

Rooms Table

TEXT roomName	INT user Num	TEXT current Turn	TEXT currWord	TEXT user1	TEXT user2	TEXT user3	TEXT user4	TEXT user5
------------------	--------------------	-------------------------	------------------	---------------	---------------	---------------	---------------	---------------