VIAQom -- Vincent Chi, Isaac Jon, Ahnaf Kazi, Qian Zhou
Title: Brick Breaking out of High School

Framework: Bootstrap.
Experience from bootstrap vs foundation lab was unanimously bootstrap favorable.

Roles:
Vincent Chi - Frontend development. Will assist with game development if necessary.
Isaac Jon - Database, game development if necessary
Ahnaf Kazi - Game development.
Qian Zhou - PM, misc

Overview:
We will be implementing brick breaker.
- Keep track of users high scores
- Have a money system which lets users purchase upgrades, perhaps:
    - More balls; balls with special properties
    - greater/more powerful balls
    - More lives
- Suggested: microtransactions between users
- Levels description:
    - Bricks will fall faster as score of user increases
    - Perhaps different types of bricks, some harder to break
   - Bad balls?

Component Map
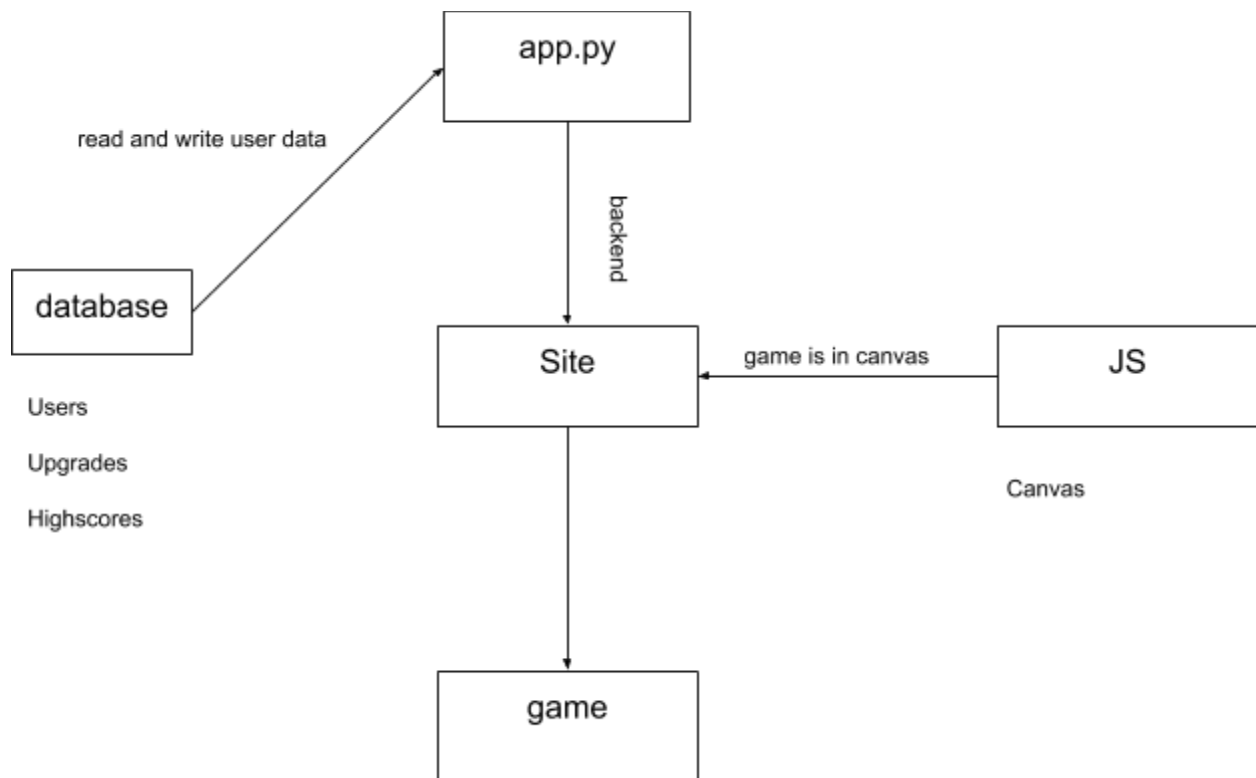
Component List:

- Sqlite: database that will store user information, to be accessed via util/ folder functions
- App.py : runs the website
- Site (ignore uppercase):
    - Template.html (if needed)
    - Landing.html (shows leaderboard, and login)
    - Login.html (or could imbed it in a dropdown)
    - Register.html
    - Profile.html (???)
    - Home.html, shows personal scores, upgrades, coins, achievements, other features if possible, levels of game to choose from
    - Store.html, shows possible upgrades, coins, (perhaps auctions made by other users)
    - Game.html, has canvas, links to js, loads current game for user to play
        - Result.html: displays result of the game, sends score into database
- JS: manages functionality of the game
    - Manages canvas
    - Needs: upgrades of given user, level difficulty

- If multiple levels, may wish to store info in a csv or json
- Outputs result into result

Site Map
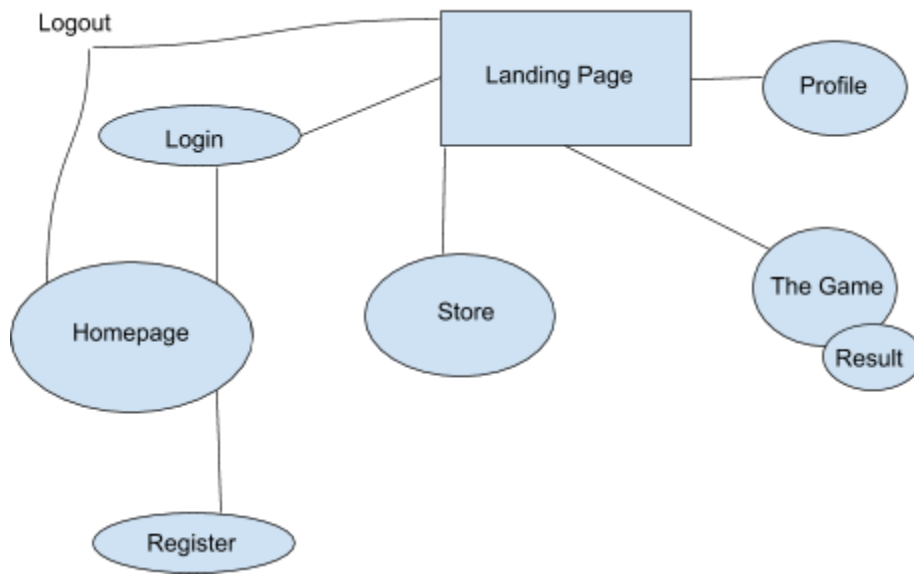
Database Schema

USERS

username	TEXT
password	TEXT
displayname	TEXT
coins	INTEGER

UPGRADES

username	TEXT
upgrade	TEXT
tier	INTEGER

HIGHSCORES

username	TEXT
score	INTEGER

Deadlines

Phase 1: starting
Due Monday, May 13:
- Devlog review/upgrade
- Database creation, database registration (add to users) and login authentication features; addition and retrieval functions for upgrades and highscores.
    - create()
    - authenticate(user, pass)
    - register(user, pass)
    - etc.
- Homepage and login/register pages
    - Also Basic templates for other pages
- Start html with game in it (write out game.html, lead to js)

Due Wednesday, May 22:
- Basic functioning game
- Linking user database features with frontend
- Brainstorming upgrade features due Friday

---- APs should be over -----
Due Saturday May 24::
- Start Score, coins, upgrades, level, etc. features
- Functioning non-game features

Phase 2: Game development
Due Tuesday, May 27:
- Each member submit implementation of at least one upgrade feature

Due June something:
- Review and integration of each others' features
- Thinking of Implementation of other potential features, start if possible