

# **UNTITLED CIRCLE GAME**

The Marvelous Moles: Joshua Kloefer, Michael Borczuk, Daniel Sooknanan, Lia Nelson

SoftDev

P04 – Agar.io Design Document

2022-06-09

Time spent: 2.3 hours

(On design document)

## **Project Description:**

We plan to create a game that mimics Agar.io, an endless game about growing bigger by eating other players.

Players will be able to play on a multiplayer server with other players and eat their friends.

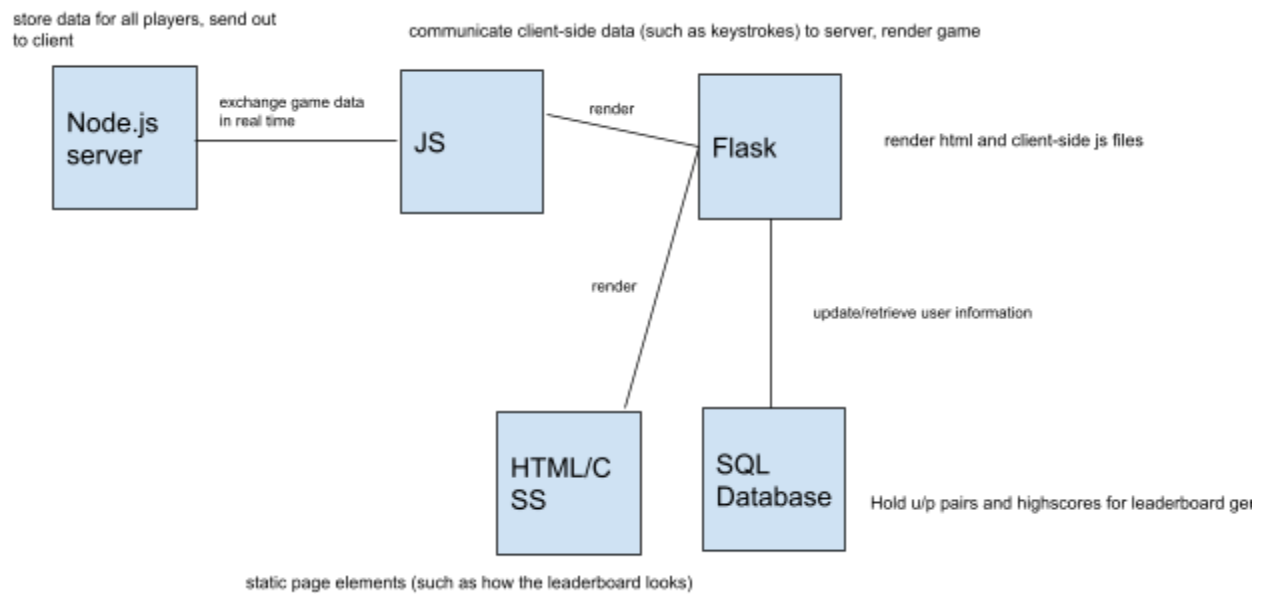
## **Target Ship Date:**

6/13/2022

## **Program Components**

- Node.js
  - We'll use Node.js to host our funny little web game and as the main means of communication between the user and the server
- Express.js
  - We'll be using Express.js to have a couple additional features to work with while using Node.js, which'll make our lives a tad bit easier.
- Socket.io
  - Allows us to emit messages and establish a two-way communication between the server and users
- Embedded Javascript Templates (EJS)
  - Similar to Jinja, EJS will let us utilize HTML templates while using Node

## Component Map



## Repo Structure

```
app/  
  __init__.py  
  static/  
    css/  
    js/  
  templates/  
design.pdf  
devlog.txt  
flag.jpg  
README.md  
requirements.txt
```

package-lock.json  
package.json

We believe that using Node.js is efficient due to its ability for easy client-server communication. However, Node.js by itself is insufficient - we need something to render the HTML pages and client-side JS files. This is why we are using Flask in addition to Node.js. For now, the Flask app is in a standard /app folder and the Node.js app is in the root. The pack-lock.json and package.json files play a similar role to requirements.txt in Flask/Apache.

In addition, the droplet currently uses Apache to host the Flask app and NGINX to host the Node.js server. NGINX is necessary for the use of websockets (which facilitate client-server communication).

## Database Organization

SQL

Username (text)	Password (text)

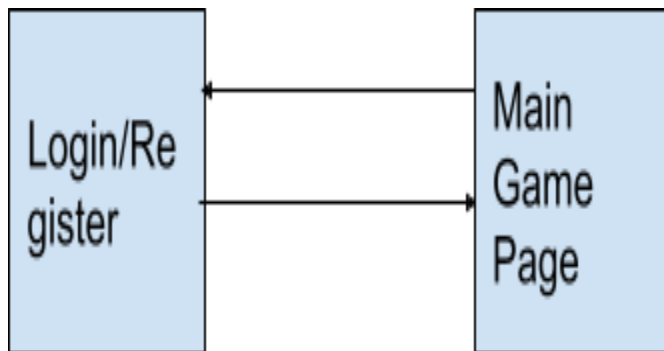
User Data (Dictionary Served) ← facilitate potential split/ejection

user: (string)	Array of arrays		
	xcoor (number, float)	ycoor (number, float)	mass (number, float)

Agar (Array of arrays)

xcoor	ycoor	mass

### Site Map



### Roles

- Lia - game mechanics (JS), backend
  - js
  - Keeping track of coordinates and movement
  - Determining when/how interaction occur
- Michael - multiplayer (JS)
  - Setting up client-server connection
  - Making sure game mechanics work in multiplayer
  - Also working on certain game mechanics
- Daniel - front-end (HTML/CSS/SQL)
  - Login/Register page
  - Assets for game
- Joshua - database integration, game mechanics (JS, Flask/Apache)
  - Assist with game mechanics
  - Implement app with Flask/Apache

### Game Mechanics

- Press the arrow keys to move your blob/cell. Holding down the arrow key causes the cell to move every millisecond. Move speed is inversely proportional to the mass of the cell.

- A cell at most 90% of your size is consumable. You will receive that 90% upon consumption.
- Blobs will start with a mass of 10.
- New agar will spawn every second.

<https://www.digitaltrends.com/gaming/agario-game-guide/>

<https://agario.fandom.com/wiki/Virus>