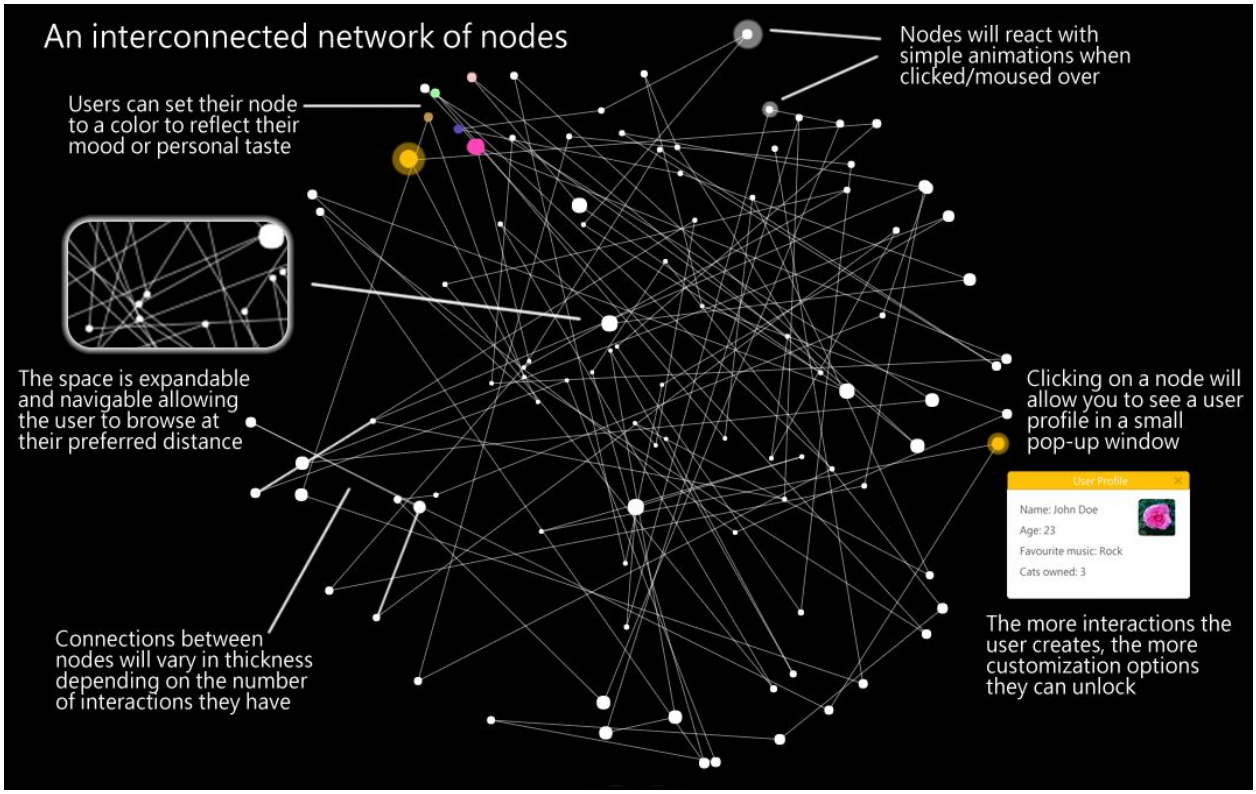
[Final Project]

[Proposal and preliminary research]

[Screen grab of Space in Action]



A Small Universe

- Track and identify users (Cookies? IP?)
- Create base nodes
- Track clicks
- Interact with clicks (lines > circles)
- Customization
- Aesthetics

Detailed plan:

- = Aesthetic/functions of secondary importance

On page load, using PHP, get and store each new user's IP address in a database. On every consequent page load check for IP again and search through database for a match.

If no match is found, create a new node with a unique ID at a unique location on the canvas

- Nodes need a buffer distance so they don't overlap

If a match is found, have the user attention brought to their node (Zoom and center, a glow, something).
Nodes should be created and ID'd using an array that adds a object when a new IP is detected.

Once nodes are set up properly, we need to create the links between them.
Once a user has been given a node and an ID, the ID can be used to find node coordinates and draw lines from one to another.
Set up a click counter using IDs as well and use that to detect interactions. This will be used to change link width and opacity to represent the strength of a bond between two nodes.

- As the link grows stronger, have the nodes drift together. Links with more clicks take priority/move less.

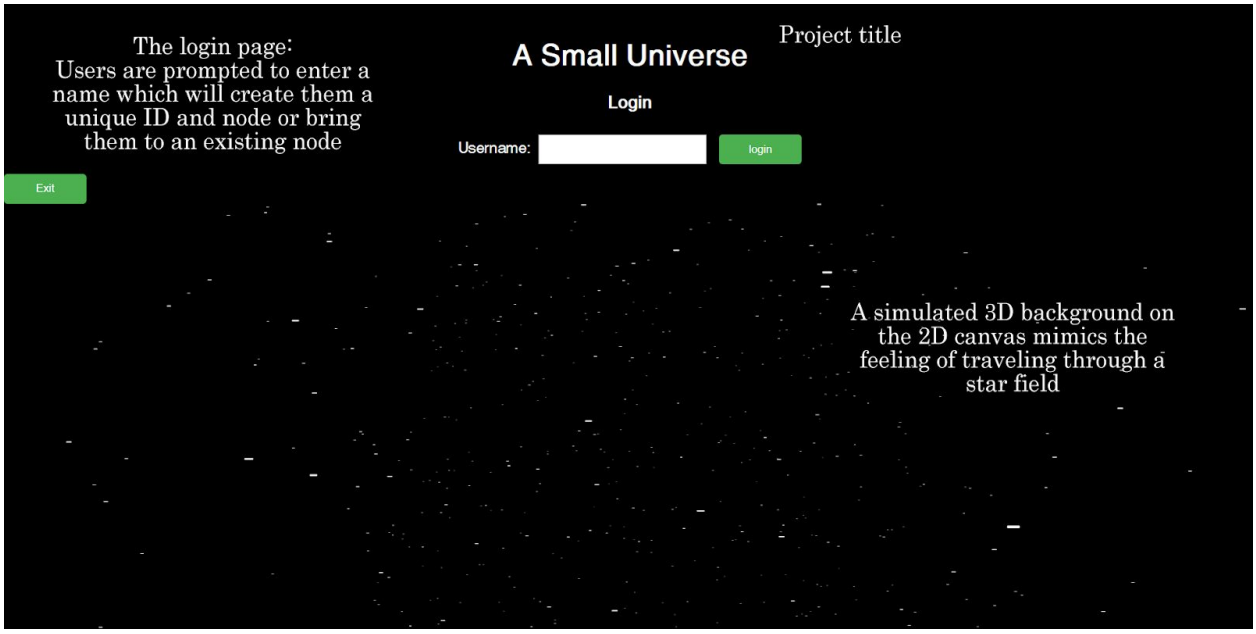
Once a click tracking system has been established, use that value to enlarge/shrink nodes according to "popularity".

- Add a timer? So actions made outside of a certain period (e.g. 24h) are no longer valid.
- Have motion feedback on hovers and clicks, small animations such as blinks, changes in opacity and size, color, etc.
- Have the canvas be zoomable and pannable

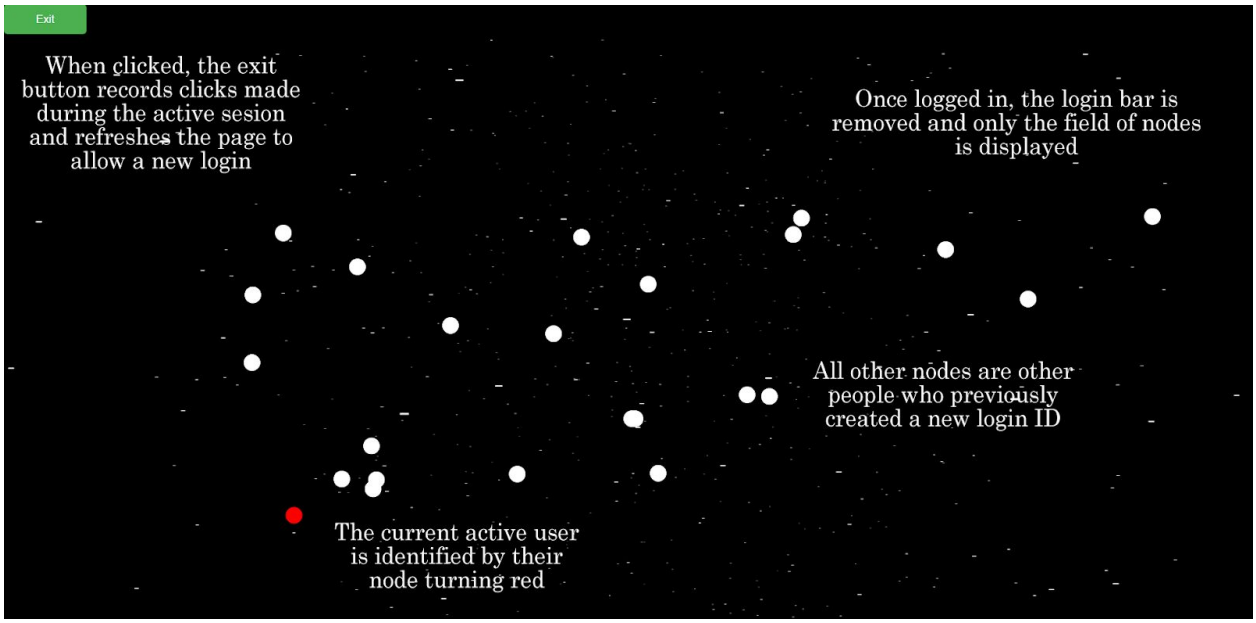
Customization:

- Give users the ability to choose colors and develop themselves a profile of sorts
- Adjust overall aesthetic of the project according to inspirational sources

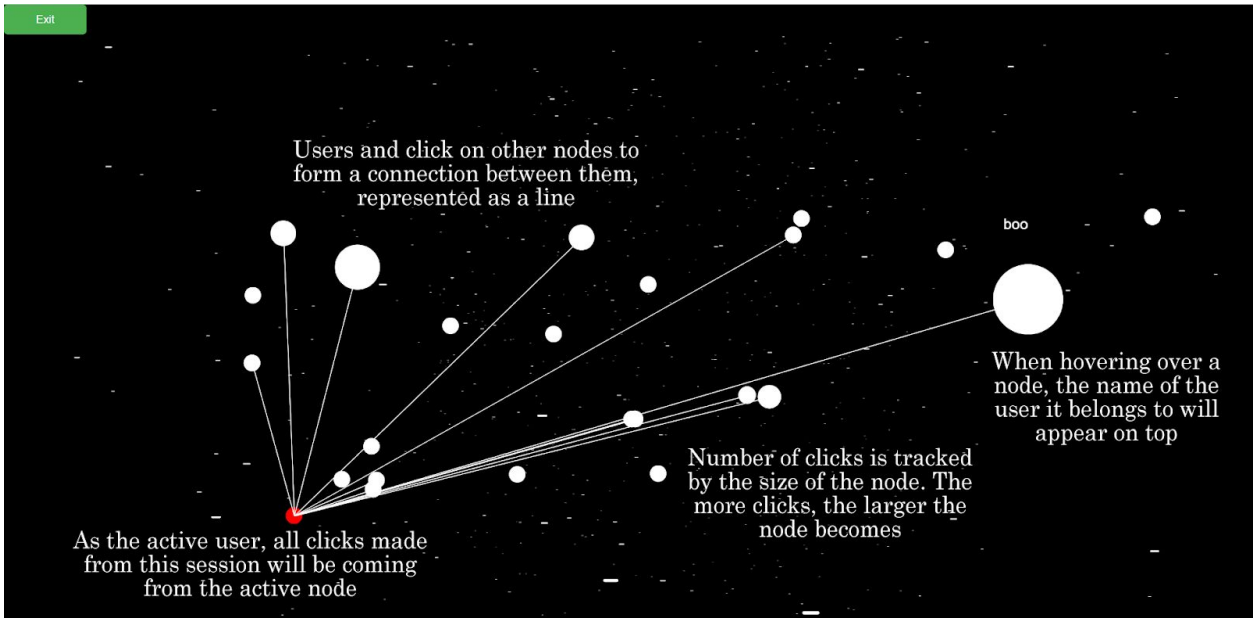
Login page:



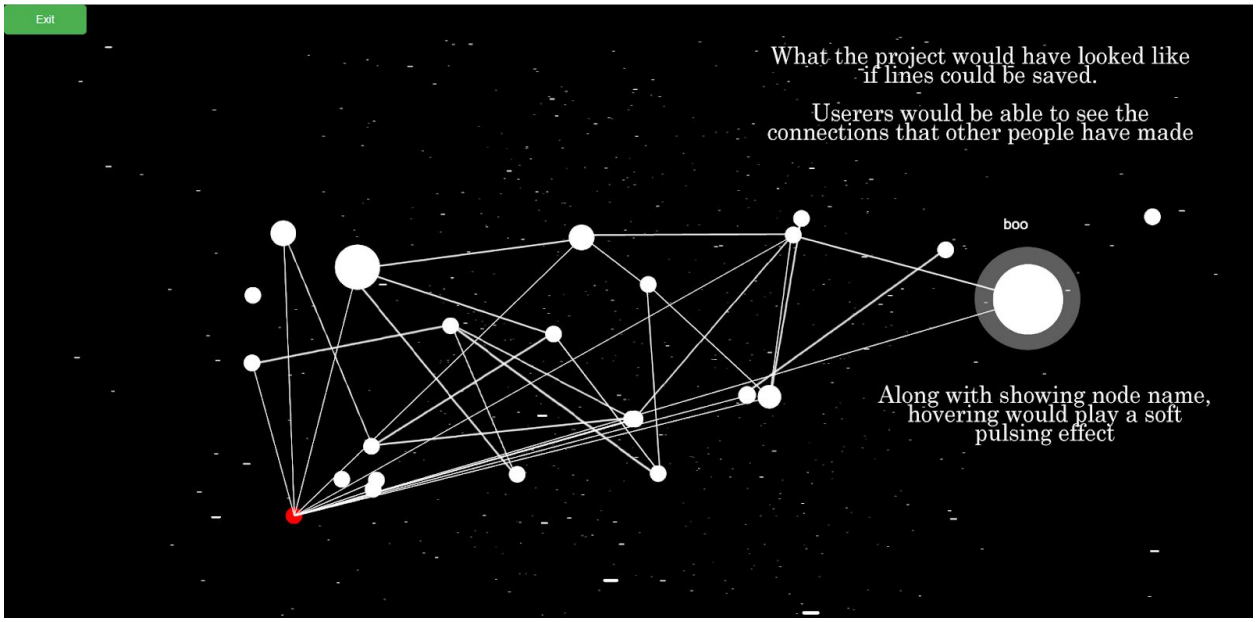
On login:



User activity:

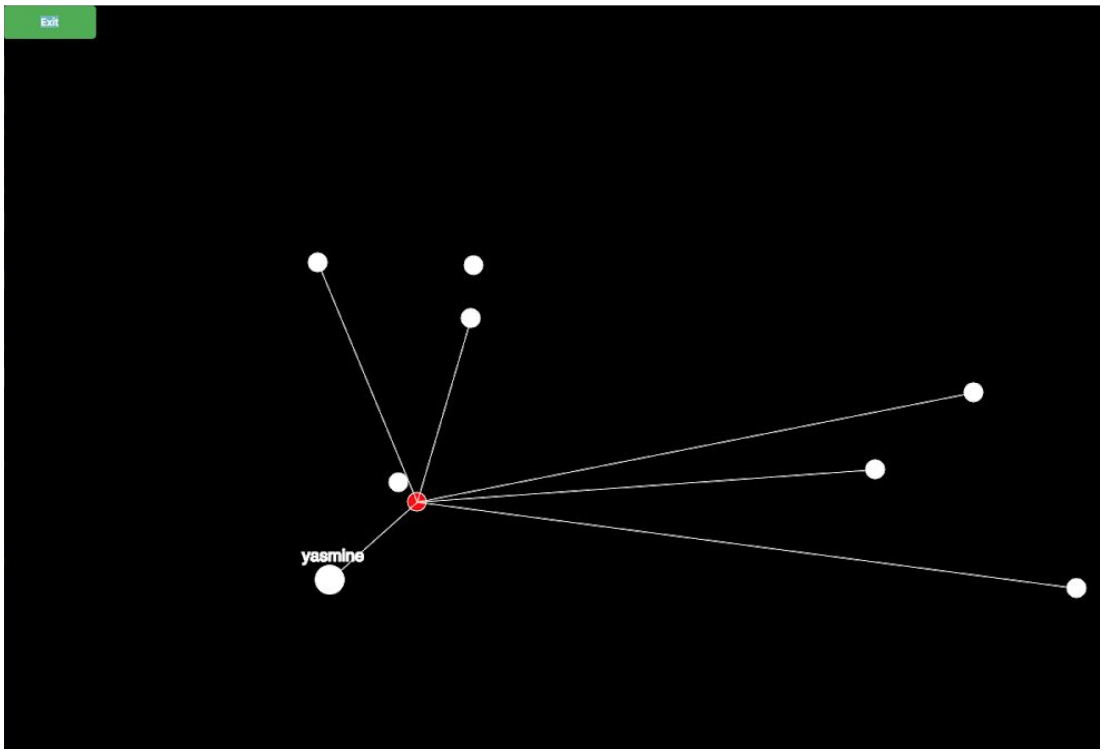


If things worked out :")

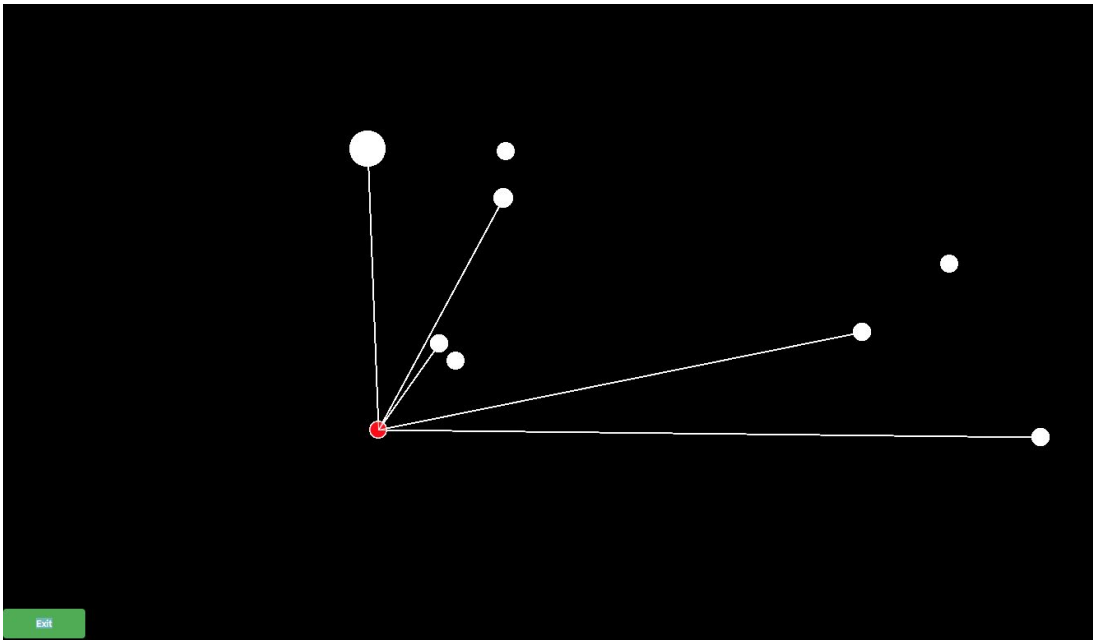


Previous iterations

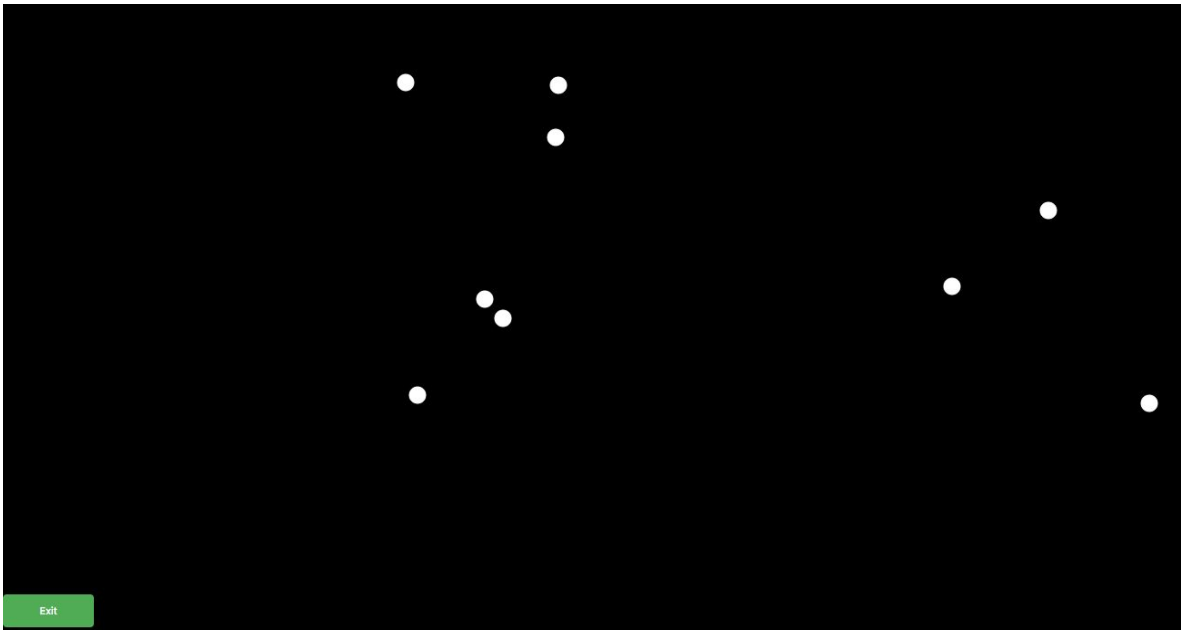
Here the project is mostly functional but missing the background and the name does not expand with the node, causing them to overlap if it grows big enough. Clicking the exit button refreshes the page now, effectively ‘exiting’ the user.



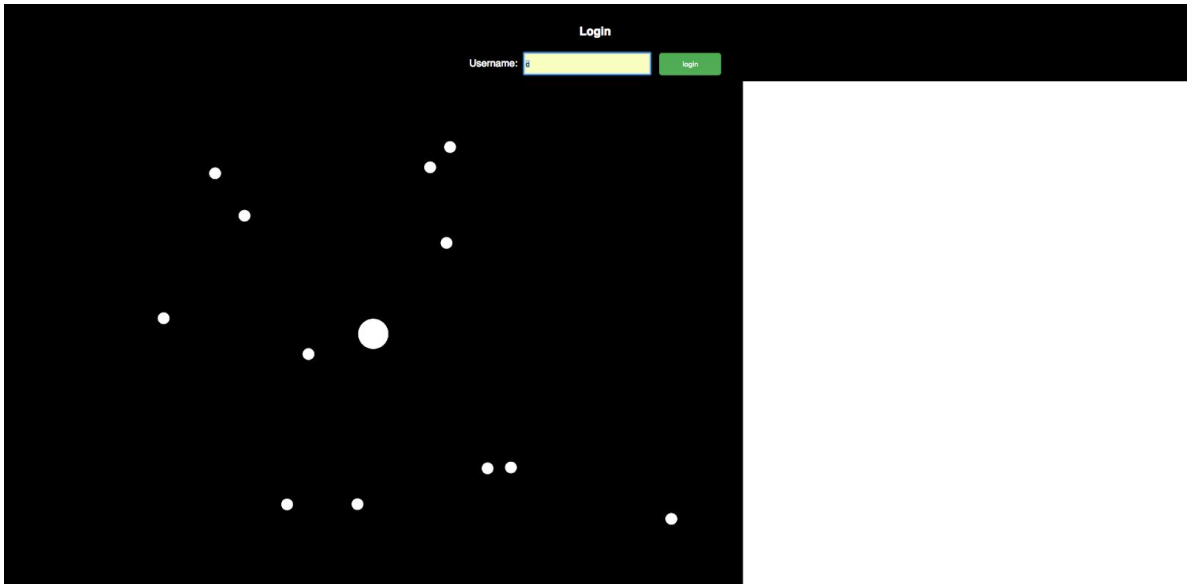
Nodes are working and the current user is identifiable but no names on hover.



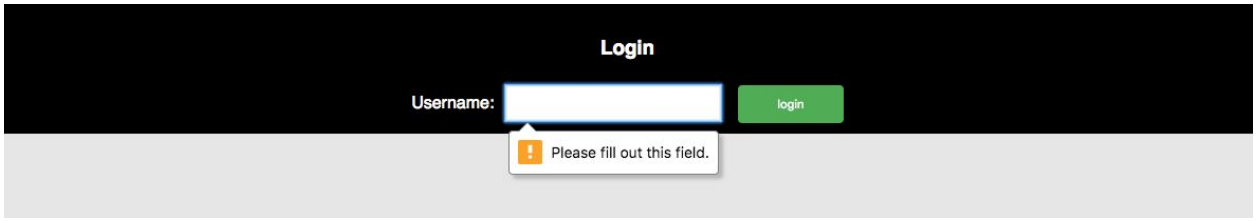
Node positions are fixed to users now and remain constant. Nodes do not appear to interact with clicks and connections have not yet been implemented. The login bar now disappears on login and both it and the canvas are more responsive to window size changing. Removed centered test node. The addition of the exit button creates a database for arrays containing the current user and clicked user coordinates. Clicking the button saves all clicks made during the active session.



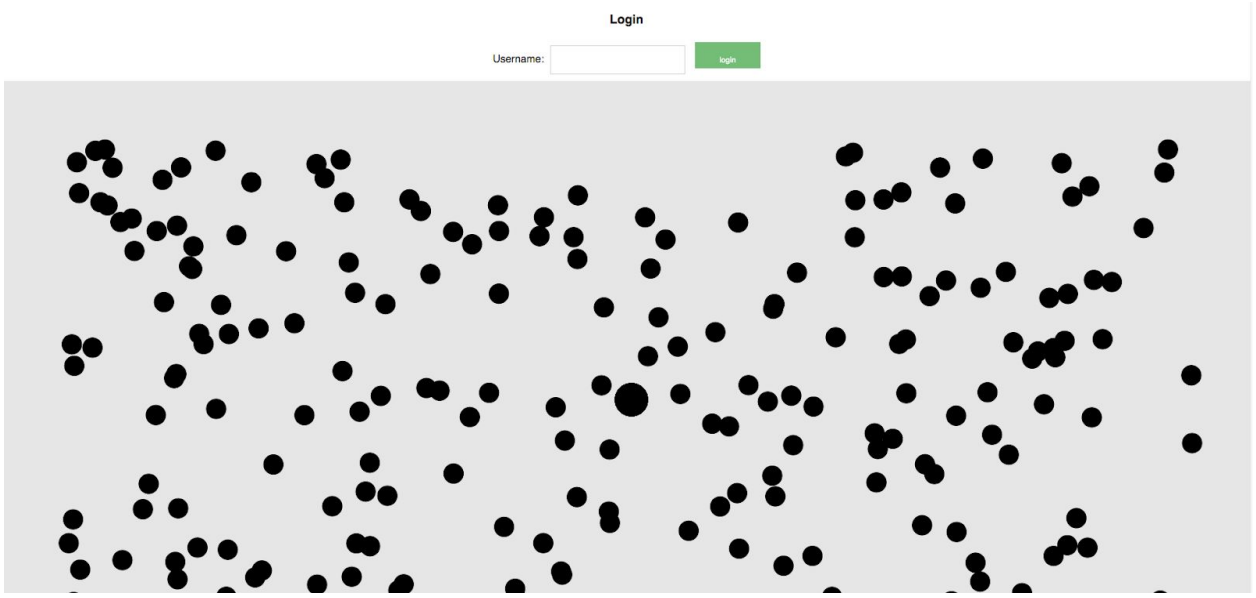
More styling fixes. The number of nodes being created at this point are limited to the number of users but appear in random positions that change every login. At this point we realized using the IP address was not necessary and people could be assigned a node using unique usernames for an ID instead.



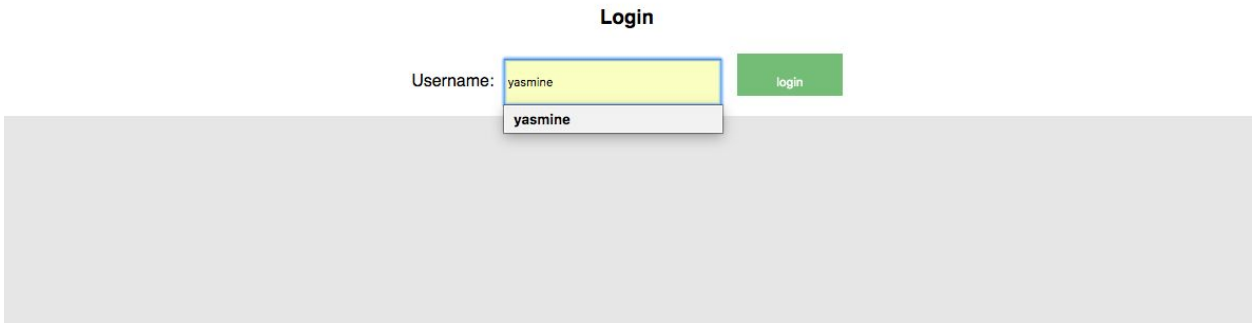
Previous login bar after a few styling changes. Moving closer to the intended aesthetic. Users can no longer submit an empty username and “Break” the site.



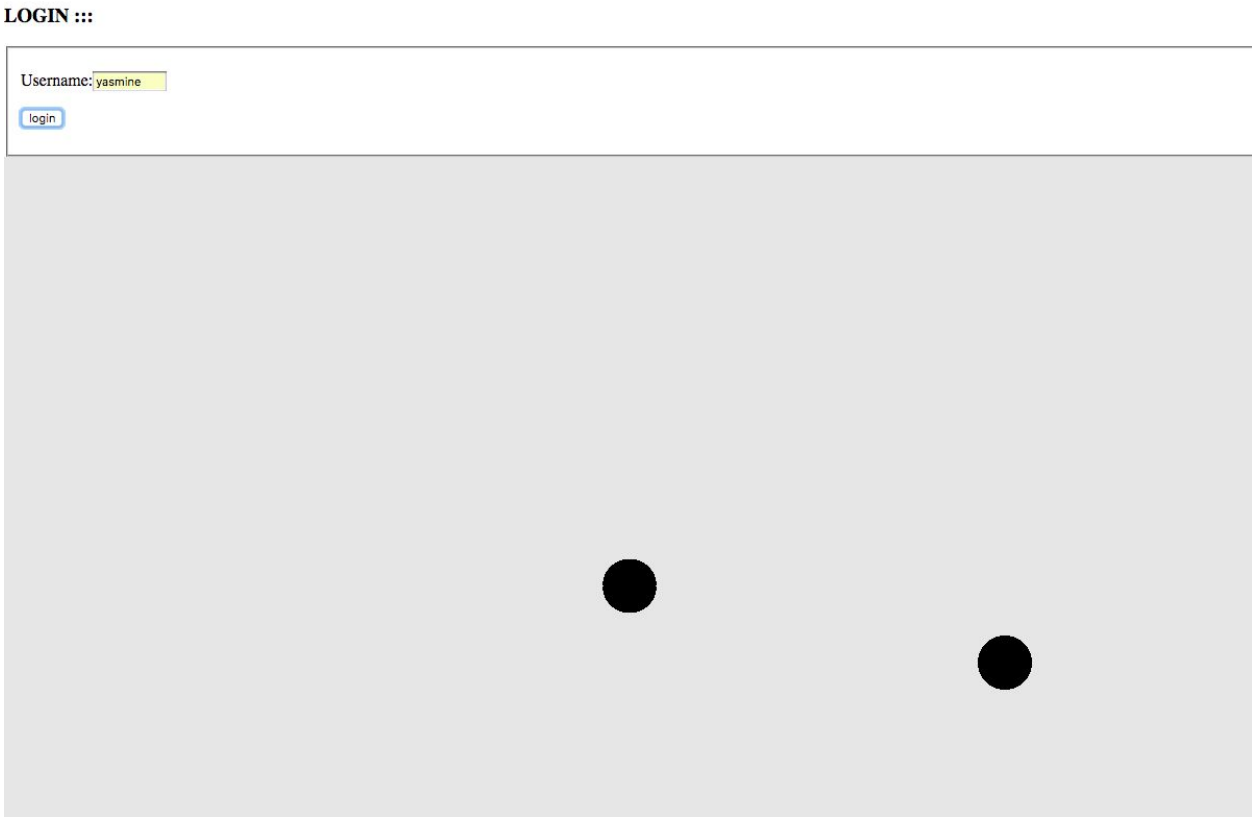
Login bar did not disappear on user submitting a name. Instead of creating one node, there was nothing linking created nodes to the user. On login, nodes would be drawn on the screen until the limit for the number of nodes was reached. The large center node was kept for testing purposes, to see if things were being drawn at all.



First working login bar. On submitting a name, the bar would disappear and the large node from the previous iteration would be drawn.



Very first iteration of the project. Included a very basic login bar that didn't serve any purposes yet and the canvas would just draw a node in the center of the screen. Resizing the screen drew more nodes in the new center.



- In creating our project, we explored different means of storing data in real time to quickly access it. We decided to use PHP in combination with MySQL, and use JSON and Ajax as intermediaries between the SQL database and the core code in HTML and javascript. We defined the nodes as special shapes that carried specific parameters that would be updated or created as new users were made. Therefore, any parameter that was affected needed to be turned into a JSON object, and either passed and interpreted in PHP or in javascript. Anything related to the user would have to be recorded within the database, which required extra measures to get that data transferred.