The Dundies Arcade

Hunter Dacey
Austin DeBoer
Gabriel Korkmaz
CIS 371 01

<u>**Semester Project**</u>

Problem Definition

- What is the problem?
  - When doing research for this assignment our group was not able to find a game arcade in VueJS. We found games that could be written in vanilla JavaScript but none that were easily playable and in one central location. Given the knowledge my team gained from the Vue section of this class, we decided to take on this project. We knew that this was not going to be easy as there were not many sources we could rely on, but our ambition directed us to accept the challenge and learn how to create this game in VueJS.

- Why is it important and/or useful?
  - Researching games developed in VueJS is not easy. There are not many to choose from and they can often cause the user to leave the site do to the lack of time spent in user experience and simplicity. Our site implements games that have been written in JS and converts them to use VueJS. Additionally we have put an emphasis on user accessibility and simplicity when creating the design and layout of the web app.

    Additionally, as the webapp grows in popularity we can add ads to the page and gain passive revenue from it. This would allow for us to have an additional income for little time to development due to the future-proof framework we built for the webapp.

- Scope
  - In order to achieve our game cabinet, we first have to create an arcade themed environment. To do this we directed our attention to how arcade machines are designed. We formatted the webapp with retro fonts found in arcade systems today and previously. We then added a border to our webapp mimicking the structure of an arcade machine. Our game selection includes some of the most classic style games, something we felt everyone would enjoy when using the site.

Hunter Dacey
Austin DeBoer
Gabriel Korkmaz
CIS 371 01

- Requirements
  - In order to comply with the project requirements for this class we created a login page and connected Google Login with Firebase. Additionally we added a leaderboard so players can compete for high scores against one another. To personalize a page, a user has the ability to change their profile pictures and their names.

Solution

- Sources of data
  - The sources of data are:
    - Names
    - Games
    - Scores
- Steps taken to load the data in the database
  - The data is gathered from a simple push call. If the score is greater than 0 then the data gets pushed to Firebase with AppAUTH user information as well as the game information.
  - Additionally, we add a call to the database that grabs all score information and then loads the data into their respective simple tables.
    - The users username is the section of characters from their email address prior to the "@" by default however the user can choose to change this.
    - Each user cannot enter the arcade without login credentials. This was done similarly to how HW 05 was completed. After creation of the account the user has the ability to play any of the games, change their profile name, and picture.
    - When playing the games the users scores are recorded and entered into the Firebase database and then displayed on the leaderboard page of the webapp. As the user plays the game their score gets added to the leaderboard should they have beat their previous score or another score.

- Alternative Solutions
  - An alternative solution to a Firebase database is Firehose. We however did not choose to use this for the reasons detailed below.

Hunter Dacey
Austin DeBoer
Gabriel Korkmaz
CIS 371 01

- ■ Advantages
  - ● RESTful design principles.
  - ● No need to parse complex protocols.
  - ● If the application was already written, then Firehose would not force you to rewrite the application over again.
  - ● Rely on WebSockets and sub server code that works in tandem with client-side JS models.

- ■ Disadvantages
  - ● Unfamiliarity.
  - ● Not as widely used.
  - ● Not as widely accepted.
  - ● Not the "standard" when it comes to storing data.
  - ● Unsure of Google Authentication is supported.

- ● Challenges
  - ○ We had a lot of challenges with converting the JS games to VueJS. Additionally we had challenges with styling and centering button text. Finally we were not able to format all canvas properties to their desired width and height to center the games with the designed gray area.

Future Work

- ● Potential Improvements
  - ○ Messaging Serve
    - ■ if we were not restricted with time we would include a messaging service so that players can communicate and compete for high scores.
  - ○ Transitions
    - ■ We would include more animated transitions and look into creating unique transitions for each game.
  - ○ More Games
    - ■ Adding more games into the arcade cabinet.

Hunter Dacey
Austin DeBoer
Gabriel Korkmaz
CIS 371 01

- ○ Game Audio
  - ■ Each one of our games can have audio attached to the experience but we did not have time to properly implement this feature.
- ○ Styling
  - ■ Given more time we would better center the games to the middle box that you see throughout the page.
- ○ Buttons
  - ■ Centering the button based off the size of the button instead of being slightly to the top.

- ● Limitations
  - ○ Knowledge
    - ■ We learned a lot during the semester and during this group project. However in order to fully launch the application on the internet we need the polish the UI and make sure that everything is correctly centered and better suited for deployment.
  - ○ Time
    - ■ This semester project was limited to the time that it was given to us. With prior knowledge or a full year course we are confident that we would be able to polish the UI and add more functionality to the webapp.
  - ○ Accessibility
    - ■ We are not sure how the webapp will react to multiple users at once when there are more than 3 of us. Given more time and knowledge, we would want to focus a portion of our efforts in pen testing to simulate how a more realistic environment would cause the webapp to react.
    - ■ According to Firebase we will be allowed 100 simultaneous users at once.