

- **What features were completed?**

Game 1: score keeping, random object generation, collision, control, math input.

Game 2: timer, random number generator, level score (updates after each problem) and game score (updates after player moves on from level), score checker that won't let player move on until they get a certain number of points per level, different number of points assigned depending on equation and level

Game3: timer, function to generate question in range, game score, check if answer was right, start game

- **What worked during the demo?**

Game 1: full playthrough and score received.

Game 2: The full game: question generator, timer, all three levels and the difference in difficulty, score keeper (both level and game)

Game3: timer, function to generate question in range, game score, check if answer was right, start game

- **What issues were faced either during the development or during the demo?**

Game 1: Holding left also moves the input to the left, leading to sometimes typing the answer backwards.

Game 2: Threshold for each level would be easy to hack as it is just a specific number times the level in our javascript

Game3: Some code was not pulled before pushed so some code got lost and had to be recovered. Some Javascript wasn't working properly so we had to scrap a lot of work.

- **What were the suggestions offered by the TA?**

To use a database for variables, questions, and more in order to secure the website against security threats.

For game two it was suggested to move the question generation to the back end because with it being on the front end, it is easy to hack and manipulate scores and the questions that are going to be displayed to the user. He also suggested that we do

some more research on the back end of things and gave us pointers on how to get started as we were a bit confused about that.

- **Individual contributions by each team member**

Kylie- GAME 2: html, checking answer, looping through problems during the level, keeping track of score, switching levels

Duke- Created underlying code for 2D boat game, collision, movement. Scorekeeping, random obstacle generation.

Riley- CSS: Build a basic style sheet that will be implemented in the final design.
GAME 3: Made the entire Javascript for game 3. Functions/features include: creating problems, timer function, and generating problems, checking if problems are right, levels and startgame.

Jules- GAME 2 specifically: Problem generator function, random number generator, displaying the numbers on the screen

Davis- DATABASE: Set up the database that will be used for our website (have yet to connect it to our website). GAME 3: Worked on the css styling, creating problems, and making the game responsive to user's answers.

Chelsea - Added functions to 2D boat game as well as math implementation and graphics. Created login page for website with graphics and design included