Meeting 1 (in class, Thurs, Feb 3)

Meeting times:
11-12 Friday Peirce
Tues 9:30

Asmod used javascript & socket.io stuff to code a web-based game in software development in less than a month with a team of 3 people. So the coding should be very doable, and we'll have time to learn. We could learn bootstrap to make the interface nicer (it's a template).

Questions for teacher:

- All we know so far is that we want to ask them questions about what they want. Do you want us to run the entire class? If so, what should we be teaching them? Do you want us to teach the class actual coding or just the process of software design? Testing software and giving feedback won't take an entire class period.
- Is there anything in particular you want the game itself to help students learn?

Questions for students:

- What do you guys want to see in a game? What kind of games do you like? Sports?
- What do you want this game to help you learn?
- Things it could teach: vocab words, spelling, science simulation

We seem to be collaborating well without a set leader for now. We'll see if we need one later. Alex will be the recorder. Asmod will be the Git Manager because he has previous experience with Git. Alex will take the lead on prioritizing teaching the class if that's something the teacher wants us to focus on.

Meeting 2 (Tues 2/8)

School:

ask them to draw on the board and talk through what they think a user would do in the program

leaderboard with top person total, of week, and of day

Saurav made a couple quick website games. The code is very short. So this will be doable in a week or two.

Talk to teachers and ask if there's something the students are struggling to learn, something they need to practice more.

Topics:
- math, pattern recognition
- music
- science
- grammar

Questions & hangman-style response
Animation will probably be too difficult
Shoots and ladders game with questions

Asynchronous or versus

What kind of thing we're trying to make:
- take something that is boring to learn and make it fun. ask what they don't like learning
- similar to a cool math game in terms of complexity

Plan for class:
- Ask them what they find the most boring or miserable to learn/practice
- Ask what kinds of games they like. Have them look around coolmath.com
- Hopefully we'll narrow in on one or two ideas and be able to start planning the layout and other details

Saurav & Alex will meet outside Wiggin Street Coffee at 2:07 so we get to the school around 2:15. That will give us 20 minutes with the class.

Meeting 3 (11am Friday)

When you solve a problem, you get a letter towards a word.
Get points for each problem. Maybe you get fewer points for each subsequent problem and you get a similar bonus for guessing the word early. You can
Maybe there are only as many questions as there are letters in the word. You only get a letter if you get the question right. So if you get lots of problems wrong, you get fewer letters. Once the questions run out
, you can get a hint (definition of word), but it'll cost points.

Final point system: You get points for getting the word right, but you get points taken off for every wrong guess or hint used. You don't get points for doing problems, but if you do more problems correctly you get more letters and then can guess accurately without using hints.

Running through some kind of world. We'll have to get someone else's code.

How to include randomness?

Mini games? One per day for extra points. Work with others who are online at the moment?

Meeting 4 (Friday night)

Goals:

Outline how we want the game to work, then figure out the walkthrough animations.

Animations from here

Interface:
- fun button
- Menu / option select

Options for letter reveals:
- Squares move
- Draw block letter

Stats updates:
- Curve transition

We looked at Slither.io and kind of like the idea of running around a world like that, but **live multiplayer is going to be too hard to code**, and we want it to be more fast-paced.

2D plane you can go anywhere using arrow keys (maybe an option you can set? Arrow keys or ASDW).
- There are coins and obstacles. Obstacles can pop up randomly. Obstacles subtract coins from your total.
- There are plants around, some of which are edible and some of which are not. You gain or lose points accordingly if you hit them. You can treat them as random and then learn them over time.
- Maybe there are mazes to go through to get a higher concentration of coins.
- Monster can show up that you can shoot. It chases you.
- Once you collect 50 coins, you get a problem to solve.

1D: like the google chrome dinosaur game
- Plant idea also works here

Competition idea (asynchronous, so easier to code): Send a challenge to someone else, in a subject of your choice. You each answer the question on your own, but the system records

whether you got it right and how long it took you. Kinda like a bet. Limited to 2-3 challenges a week.

Do we need Unity? Or can we do this all in JavaScript? Ask Skon.

Alex will email science teachers to see if we can incorporate anything from that curriculum. Also history?

Log in
ID
https://www.youtube.com/watch?v=U3sT-T5bKX4&t=215s

Summary of game attributes we have so far (hopefully for a requirements doc)

- It will be an asynchronous online game.
- Students will be able to log in so their cumulative points are saved. There will be a leaderboard showing the top scorers for the day, week, and all time.
- There will be some sort of “main world” through which students can move/run and collect (and lose) coins. When a certain threshold of coins is earned, the student is given a problem to solve.
- The problem will be multiple choice or will require students to type in a number. No free response. Each problem should relate to the students’ curriculum in some way.
- If the student gets a question right, they get a random letter of a mystery word. If they get the question wrong, they do not get a letter. Either way, they are sent back to the main world to collect coins and reach the next problem.
- Scoring: For each word, there is one problem per letter. So if the mystery word has 6 letters and you solve 4 of the next 6 problems correctly, you will have 4 letters revealed before you are required to make a guess or use a hint. Guessing the word gives you a certain number of points. Each hint detracts points from that prize. Each incorrect guess also detracts points.
- There may be a “duel” option where you can challenge another player to compete with you on a more difficult problem. You each bet a number of points, and the winner gets all of them.