Meeting Minutes

19th September 2017

Goals for the next week:

- Planning out the activities and challenges (new focus as per feedback from stand-up)
- Paper prototype:
    - What will be displayed on the student screens
    - What will be displayed on the teacher screens
- Start testing out the paper prototype

Tutor/Lecturer Stand-up feedback:

- Focus more on the activities and challenges instead of the management system
- Management systems have been done before so keep this to the side
- Prototype should focus on these activities and challenges
- Focus more on the social learning aspect
- Create a system that provides a unique way to learn programming not something that can be used for just any subject.

Discussion:

Reward system:

Most users wanted the rewards to be a combination of:

- Badges → they wanted certain goals to be in place
    - Be able to see all of the badges achieved
- Points

Teacher screen (for the electronic prototype):

- Teachers → Just monitor, settings

Games

- Younger students → create a red ball
- Older → will have higher level of control e.g. create a character (however they like)
- **Maze (grade 4-5) → Divy**
    - On the big screen
    - Each student starts at a different place, students create a program to move them through the maze
    - Blocks of code, simple phrases then they have to insert
- **Puzzle/Movie (grade 3) → Beth**
  (*Movies could be displayed on user profile/class dashboard)*
    - How many pieces per student
    - Picture idea: they actually create what's on the puzzle
    - Function code, animate what's on the screen
    - Little movie? Actually creating the output

- Recipe
    - Add 2 eggs etc
- **Spaceships (grade 6) → Barney**
    - How the weapons work
    - What it looks like, max damage, etc etc
    - Battle the ships
    - Chunks of code
    - Spaceship flying around and finds a memory chip/lost script which unlocks new functionality. A for loop
    - Only program stuff from a list of programs, “items” but a list of code, up to students to put them together in an efficient way
    - Shoot function
        - Hard Code number, “for loop 10 times” etc
    - In smaller teams, everyone's trying to make the best ship they can in their own group, but there’s also big monster/space battle on the big screen
- Races
- Puzzle

<u>Team/classroom</u>

- The same games can be used for each year level and group type with just different questions
- Points

<u>Testing if it’s correct/incorrect</u>

- Output vs word for word correctness.
- Higher level can explore different ways of getting the correct output/
- Pseudocode → works best for all years as it doesn’t require focusing on a particular language but more on the logic.

<u>Prototype</u>

- Prototype the activities first
- Paper → electronic prototype / online mockup / actual website
- Barebones → focus on the things people understand the least that requires to be built physically.
- Game
    - Show for grade 3 and grade 6, extremes
- This is what’s displayed on the big screen, and on the smaller student screen