CMPT 276
Group 17

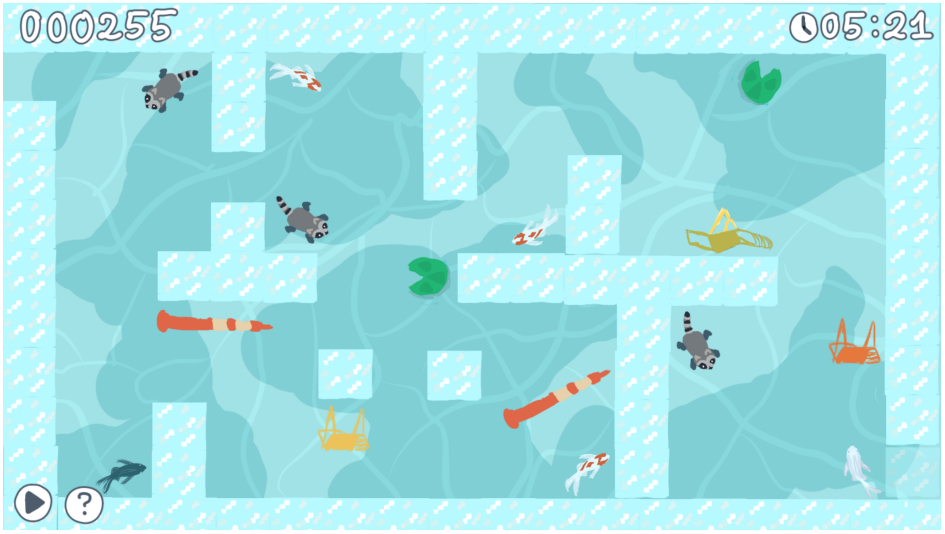
“Koi Kapers”

Phase 4: Report and Presentation

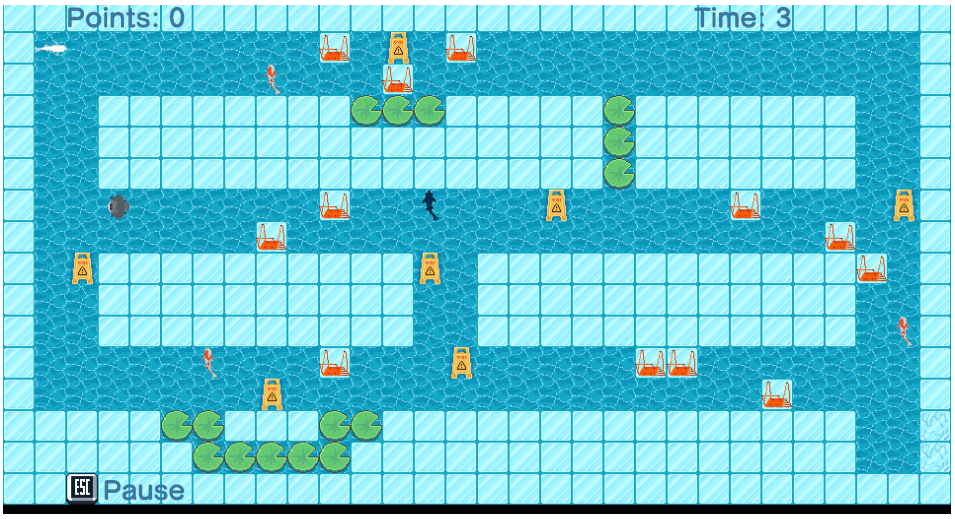
Colton Blackwell, Jonathan Bryan, Ridham Sharma, Sophia Don Tranho

The Game:

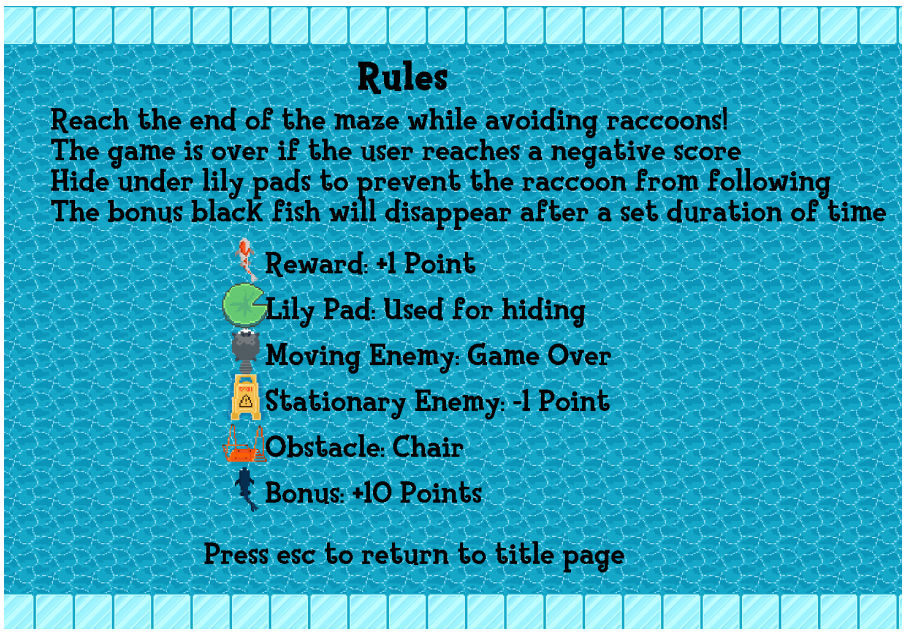
Koi Kapers is a 2D Java-implemented game that was inspired by the SFU koi pond with the pylons and chairs being tossed into it last winter. In this game, the user plays as the white koi fish attempting to escape the maze while avoiding contact with the moving enemy (swimming raccoon).



Initial Design of *Koi Kapers Vs. Raccoon Raiders*



Level 1 in *Koi Kapers* (Finished Product)



Meaningful elements within the game environment

So What Changed?

Many features were changed in the initial design compared to the final product for various reasons.

- The initial design could handle Mouse inputs → final design only supported key handling.

  We decided that it was unnecessary to add mouse input for only the menu navigation, so we kept it simple and used keyboard input only.

- Checking the rules of the game Use Case → Didn't implement the feature of checking the rules from the pause state.

  In the game's current state, we felt like the number of mechanics was small enough that the player could simply learn as they played. Furthermore, since the game is only one life, you are given the opportunity to view the main menu screen (and therefore the rules) often.

- If the user completes the level under a set time frame, the user gets bonus points.

  Rather than giving the player points for free for being fast, we had the black koi fish, which disappeared after a certain amount of time instead. This gave the added benefit of having the player need to go out of their way to earn the extra points, instead of simply being fast.

- Changed the bonus fish feature
  - "A bonus reward object may appear at a random time and random location in the level."

Rather than having the bonus fish appear at a random time (which could have caused the player to miss it entirely if they beat the level too quickly), we decided to have them visible from the start. They were also added in particularly difficult sections of the map to reach, so having them visible from the start of each level would allow for the player to plan out a route to collect them without being caught.

- Fish follow feature
    - If there are multiple fish following the main character and the main character hides under a lily pad, the tailing fish will swim to the lily pad so all of the fish will be hidden underneath the lily pad cell.

      We scrapped this idea simply due to time constraints. While we still believe it would have been a great addition, it simply was too big of a feature to add given the time that we had to complete each phase.

- *Looking at initial design*
    - If players head to the exit before collecting all the regular rewards then the game will not end.
    - It will also show a pop-up message reminding the player that they need to collect all the regular rewards.

      These two features were scrapped since we changed the way the map works. Until all the required fish are collected, the exit is displayed as cracked ice blocks. Once the fish are collected, these blocks melt and the exit to the level is opened.

    - Point system changed.

      The change here was simply the amount of points each fish would give the player. As a whole, this does not change the gameplay much, since the score is not saved to a leaderboard and each fish gives a static value in terms of points.

Lessons Learned:

- Communicating with each other often to stay on track
    - Group meetings at least once per phase
- Not everything is permanent
    - Be open and accepting to new ideas
- Playtesting is very valuable
    - Allowed us to discover bugs that our test code did not find
- Constant refactoring
    - First draft of code was often hard to read

Build Automation:

- Refer to README file in main ‘CMPT276F23_group17’ directory