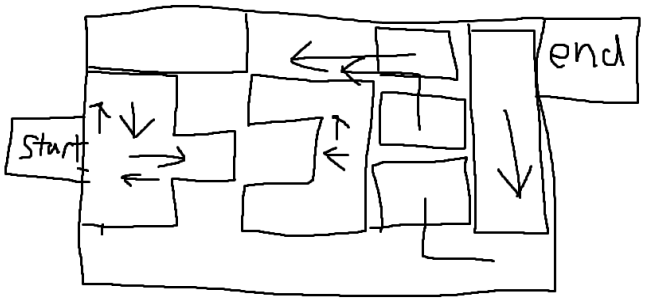
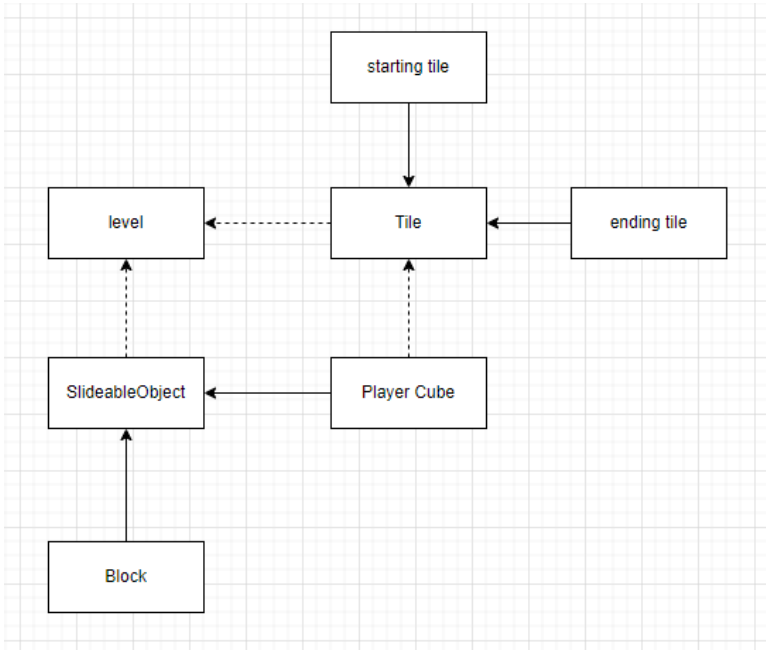
Sliding Puzzle Design Specifications

Project Description

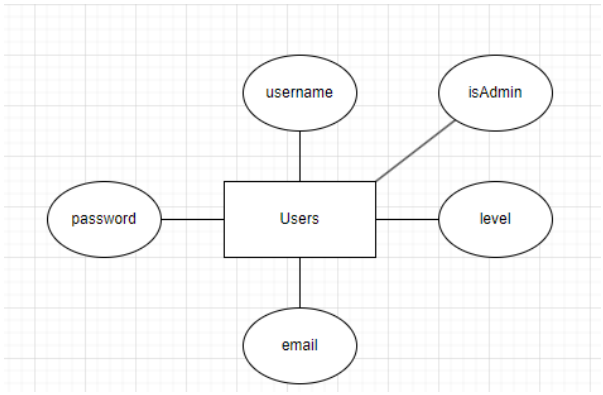
This project will be a simple sliding puzzle game where a user gets a square piece from the starting slot into the end slot. It will start simple and slowly introduce more blocks blocking the user's path. Users can play without an account, but they will need an account to save their progress. Users can sign up with their email and password and a username. There will be a single table in the database containing user information and their level.



Sketch of what a level could look like



UML diagram



ERD diagram

The Puzzle Solver

I will start with brute force and see what I can come up with to solve the puzzle as development continues.

Market Space

The market space for this application will target people who like solving puzzles and learning mechanics by doing the puzzle instead of being told the rules and mechanics.

Functional Specifications

Multiple levels: the puzzle will have multiple levels with increasing difficulty and new mechanics
Requesting hints: The user can use a hint that will move a piece or the character towards the solution.
Requesting solution: A user can request a solution to a puzzle. It will display a warning telling the user this is highly unrecommended as it spoils the fun and learning experience of the mechanics.
Creating an account: Users can create an account if they wish to save their progress
Signing in: users can sign into their account to continue where they left off.
Resetting Password: users can reset their password or have an administrator change it for them.
Removing accounts: Admins can remove accounts

Deployment

Docker Compose or Heroku

Milestones with deadlines:

- M1 (beginning of class - 2/4)
    - Brainstorm ideas for the project
- M2 (2/5-2/16)
    - Create GitHub repo
    - Create Design Specifications
    - Create a simple prototype for the sliding puzzle with 1 working level
- M3 (2/19-3/1)
    - Create a login/signup page
    - design more levels for the puzzle
- M4 (3/4-3/15)
    - Create textures for the puzzle
    - add administrative functionality
- M5 (3/18-3/29)
    - Design algorithms for solving the puzzle
    - implement hints using the algorithm
- M6 (4/8-4-19)
    - bug fixing