

# Behavior Specification

Ryan Riehl  
Daniel Sanchez  
Tristan Simon  
Mark Tsujimura

CECS 343  
Professor Hubert Huynh

### **Use case 1: Select Problem Types**

**Use-case description:** The player has to select at least one type of math problem to solve.

**Primary actor:** Player

**Precondition:** The different types of math problems are visible to the player.

**Postcondition:** A problem of the selected type is displayed.

#### **Basic Flow:**

- Player views available types of problems
- Player clicks one or more of the types to select them, putting a check in the associated checkbox
- Problem of a selected type is displayed to the player

### **Use case 2: Input Answer**

**Use-case description:** The player enters the answer to a given math problem.

**Primary actor:** Player

**Precondition:** At least one type of problem is selected by the user.

**Postcondition:** If the player solves the problem correctly, the score increases.

#### **Basic Flow:**

- Player views selected problem
- Player enters their answer to problem
  - If the answer is correct, the problem disappears and the player's score increases.
  - If the answer is incorrect, the answer is cleared and the problem remains, waiting for another input.

#### **Exceptional Flow:**

- The player enters a non-numerical answer to the problem
- The answer is interpreted as incorrect.
- The exact score increase for a correct answer is determined by what upgrades have been purchased.

### **Use case 3: Buy Buildings and Upgrades**

**Use-case description:** The player uses score to obtain more resources.

**Primary actor:** Player

**Precondition:** The player has enough points to afford a building or upgrade

**Postcondition:** The player's points are reduced and they gain a building or upgrade

**Basic Flow:**

- The player selects which building or upgrade they want to buy by clicking on the associated icon.
  - If the player has enough points to buy the selected object, then their points are reduced by the cost of the object and they gain the object they selected.
  - If the player does not have enough points to buy the selected object, nothing should happen.

**Exceptional Flow:**

- If the player successfully buys an upgrade, the upgrade will disappear from the buy list.

**Use case 4:** View Statistics

**Use-case description:** The center portion of the game turns into a page full of statistics of the game. Some examples of data shown are: Game time, total resources generated, number of buildings, total clicks, and win/loss statistics.

**Primary actor:** Player

**Precondition:** Statistics page is not shown

**Postcondition:** The player is able to see a page with statistics about their gameplay

**Basic Flow:**

- The player clicks the statistics icon on the center top part of the game
- The center portion of the screen turns into the statistics page

**Use case 5:** View Achievements

**Use-case description:** The center portion of the game turns into a page containing the achievements the player has unlocked, as well as achievements yet to be obtained.

**Primary actor:** Player

**Precondition:** Achievements page is not shown

**Postcondition:** The player is able to view the achievement page

**Basic Flow:**

- The player clicks the achievement icon in the center top part of the game
- The center portion of the screen turns into the achievements page

**Use case 6:** View Map

**Use-case description:** The center portion of the game turns into a Map where he competes with an AI.

**Primary actor:** Player

**Precondition:** Map is not shown

**Postcondition:** The player is able to see a page with the map with controlled territory by both the player and the AI.

**Basic Flow:**

- The player clicks the map icon on the center top part of the game
- The center portion of the screen turns into the map page

**Use case 7:** Attack Node

**Use-case description:** The player uses their collected resources to bring a new map node under their control

**Primary actor:** Player

**Precondition:** The map is displayed

**Postcondition:** The player's attack on the node is resolved and resources are adjusted accordingly.

**Basic Flow:**

- The player selects a player-controlled node on the map that they will use to start their attack.
- The player selects an adjacent node they do not control to attack.
- The power of the attacking node is compared to the power of the defending node.
  - If the defending node has less power than the attacking node, it becomes controlled by the player and its power is reduced.

- If the defending node has more power than the attacking node, control of that node does not change and its power is reduced relative to the power of the attacking node.
- The attacking node loses power based on the initial power of the defending node

#### Glossary of terms:

**Score:** A positive integer value that increases when the player successfully solves displayed math problems. It also increases based on buildings the player has accumulated.

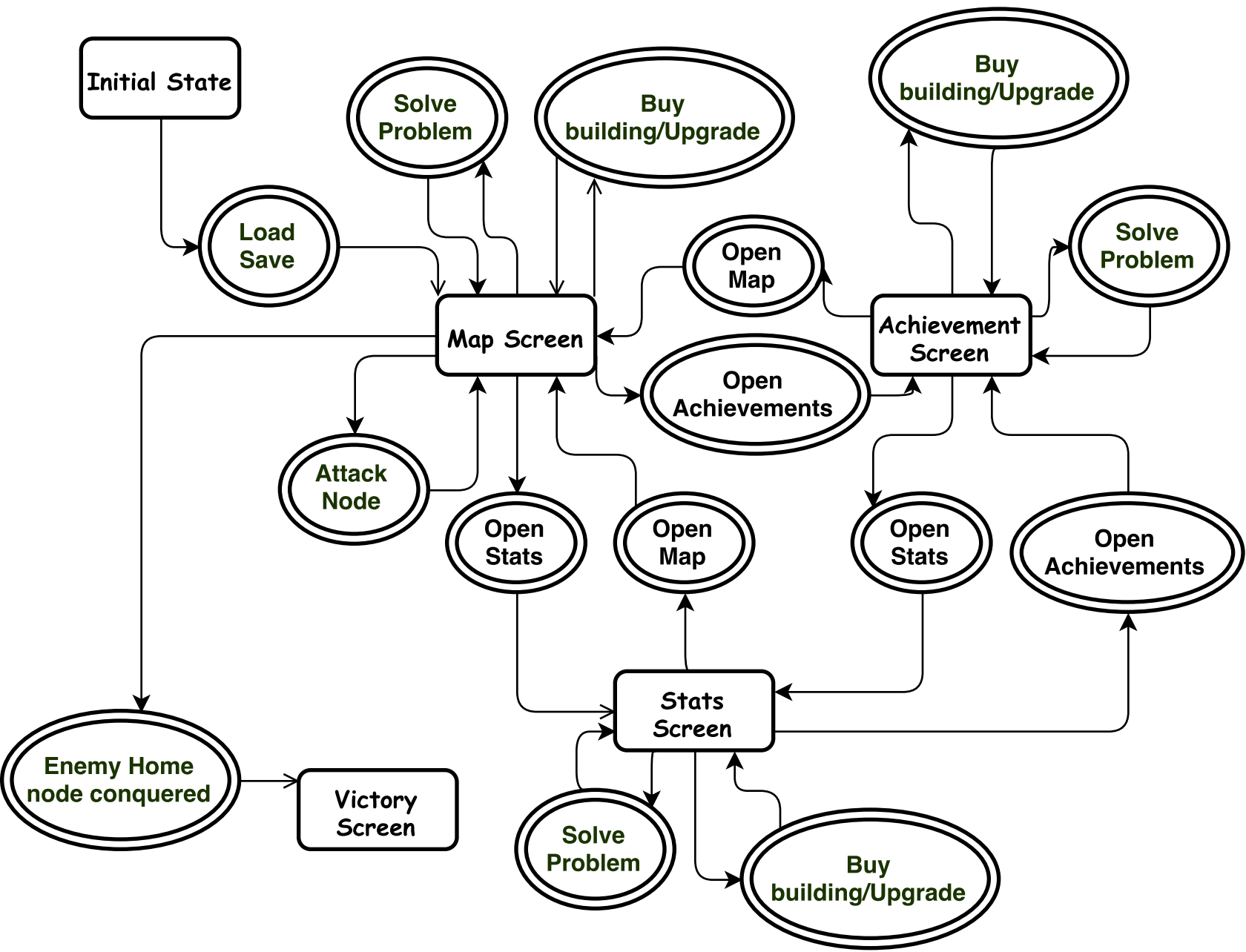
**Building:** An object that the player can obtain by spending their accumulated score. Buildings can be purchased multiple times. Buildings will gradually increase the player's score based on the number and type of buildings purchased, as well as which upgrades have been purchased.

**Upgrade:** An object that the player can spend score to obtain. Upgrades may only be purchased once. Upgrades affect various aspects of the game, but primarily affect score accumulation.

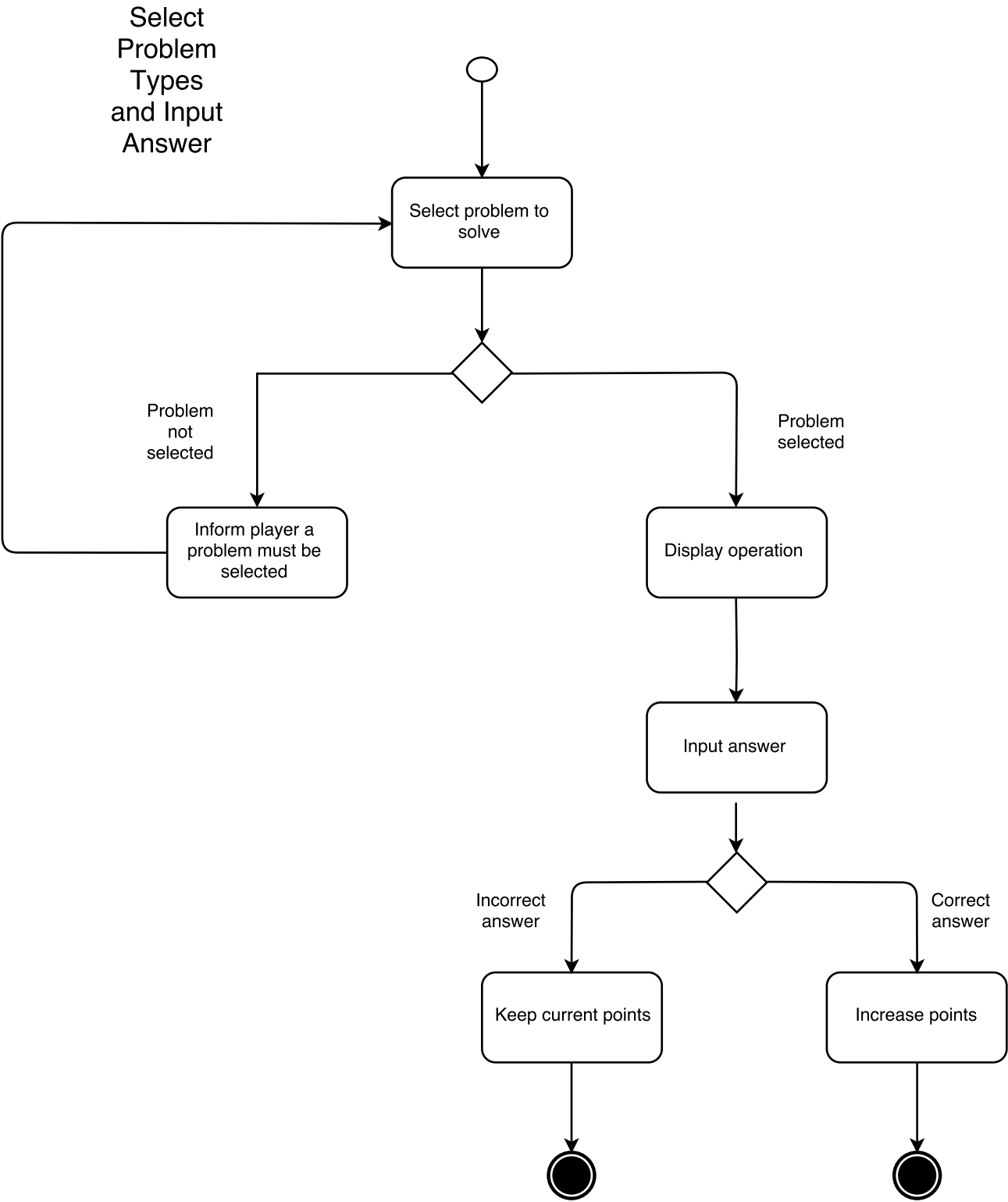
**Node:** An object represented by a colored circle on the in-game map. Nodes are either player-controlled, enemy-controlled, or neutral. Controlled nodes can be used to attack adjacent nodes to bring them under control of the attacking entity. Nodes are adjacent to Nodes connected to them by lines on the map.

**Power:** A positive value that each node has. Power is used to determine the outcome of attacks from node to node. See the "Attack Node" use case for more information.

# State Diagram

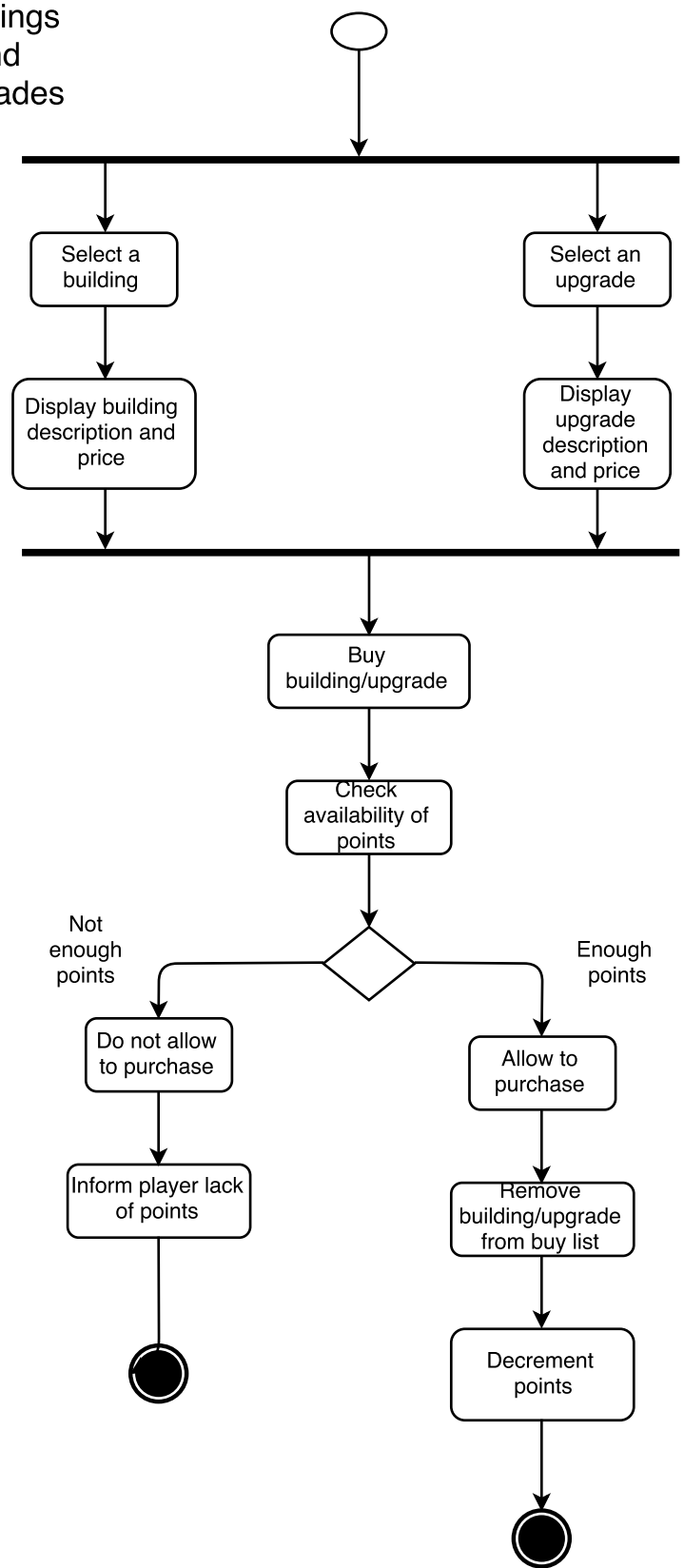


Activity Diagram  
1



Activity Diagram 2

Buy  
Buildings  
and  
Upgrades





# Activity Diagram 3

Center Portion of UI

