

Game Design Document

The game starts from the main menu Home.

A player has two game currencies stars and lights.

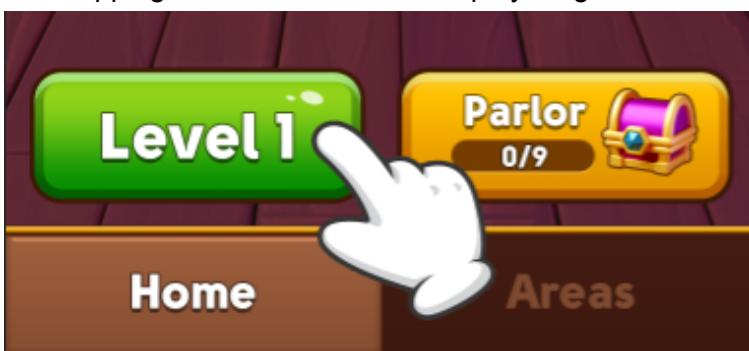
Lights - can be found in a chest on the level.

Stars - players get in the main game after completing the orders. Players can use stars to purchase objects in rooms. Objects could have different values.



At the Areas menu, players can look at all locations.

After tapping on the level button the players get into the main game.





The main gameplay is a merge aiming to get icons of the higher hierarchy.

Icons with blue lightning can spawn new icons with a double click on them. To spawn one icon, the player has to spend one light(energy).

Players can increase the amount of energy by double clicking on the energy icon, the amount of energy is settled according to config.

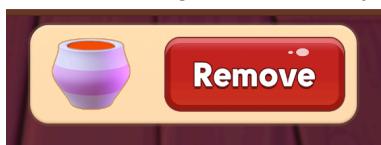
The higher the hierarchy of light(energy) icon the more lights(energy) the player would get.

After tapping on the Home button players will back to the main menu scene(Lobby)

Above the icons field, there is a panel with orders. Each order demands a certain amount of icons that should be presented on the field to complete the order, by completing the order stars accrual to the player. There are info buttons on orders, by tapping them players will see the Info popup with icon hierarchy and icon that spawns it.



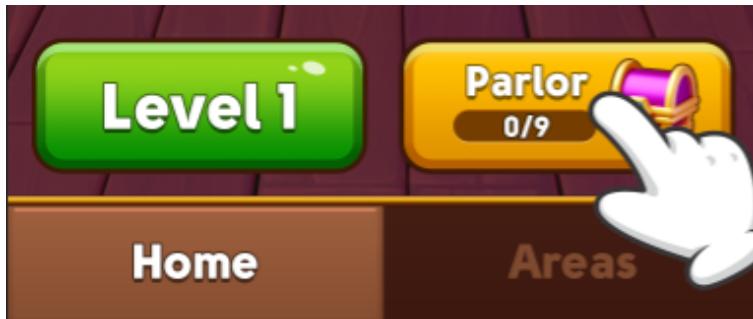
The player is able to delete any icon from the field. After tapping on any icon the panel below the icon field gives the ability to remove the icon.



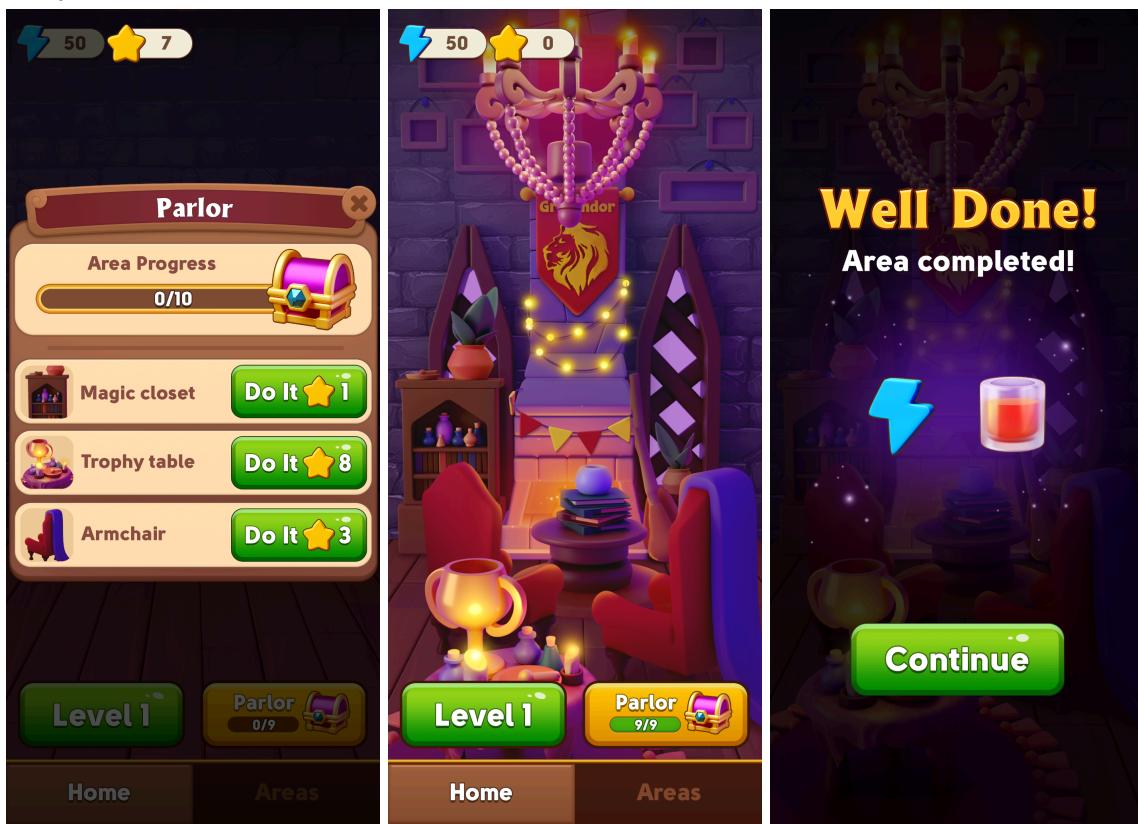
Each spawn icon (with blue light) has some spawn capacity that decreases by one each spawn. After capacity decreases to zero, the recharge timer is enabled. Timer delay settled in config.



When the player gets some amount of stars he becomes able to furnish the room with objects. The purchase objects popup arises after clicking on the yellow button in the main menu (Lobby).



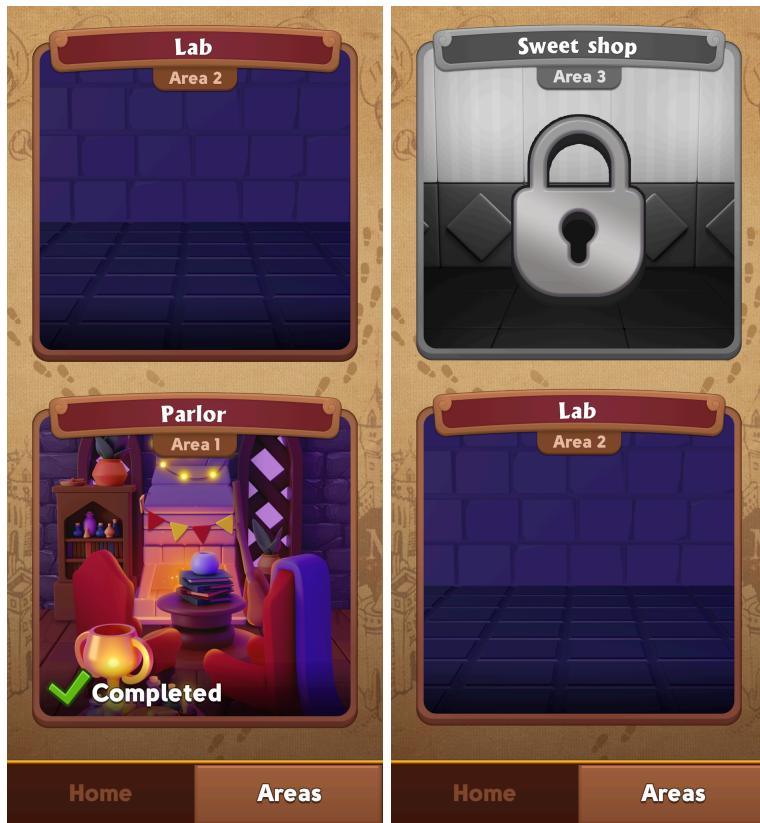
Each purchased object fills the room progress bar, after the progress bar has become full room is completed. The reward popup arises on the screen and gives some icons as a reward. The lobby switches to the next room.



After tapping on Continue the player opens a new room and new objects become available to set in.



By tapping on the Areas button the player can look at all locations. By tapping on the Areas button the player can look at all locations. Each area could be in three states, locked, inprogress, or completed.

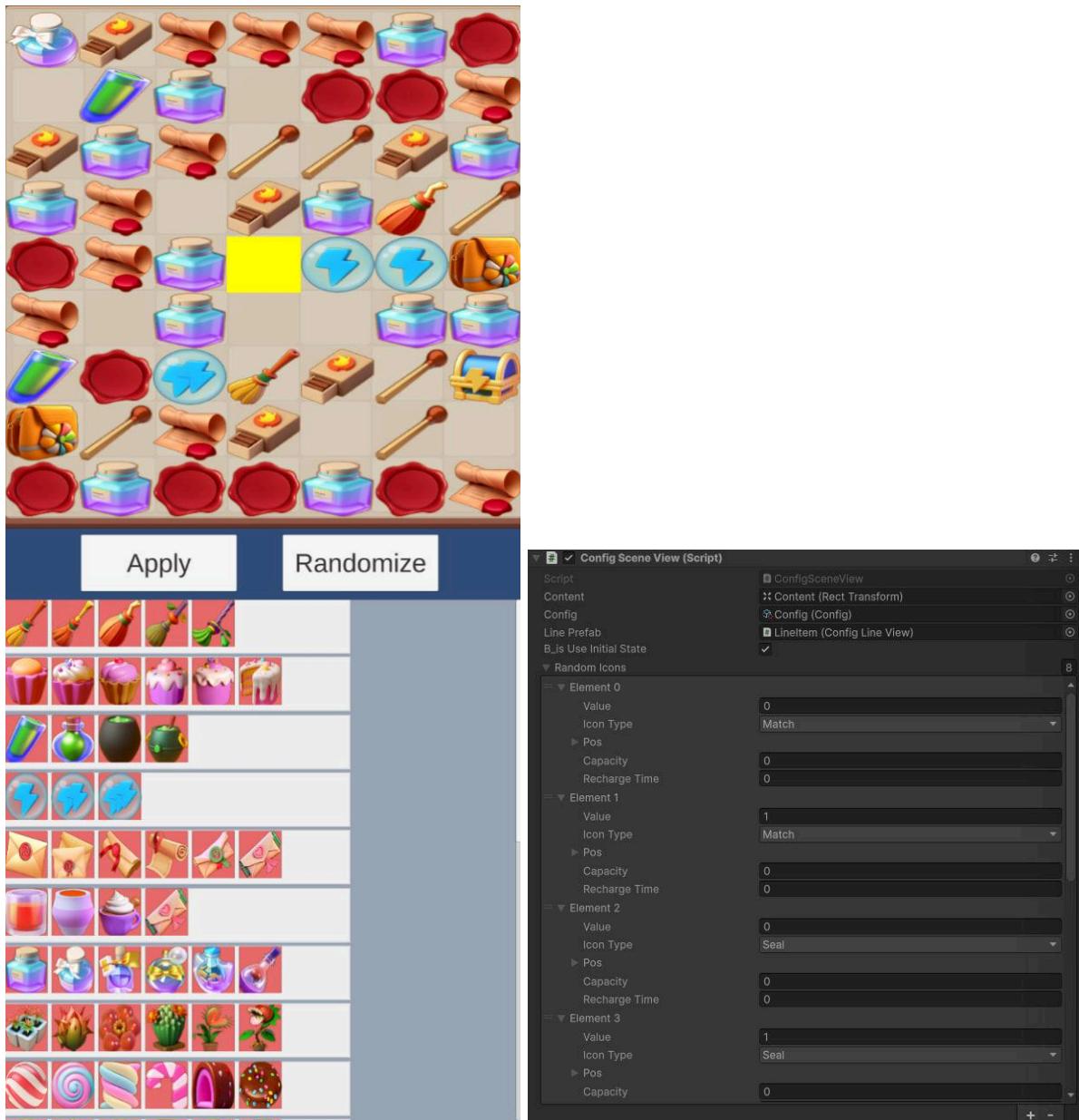


Config Scene

The game's initial field must be configured somehow. To achieve that purpose, the ConfigScene was developed. It allows the designer to set up any icon anywhere in the icon field.

A button randomize was created to speed up the process. The ConfigSceneView script designer seated the list of icons from which random events would happen.

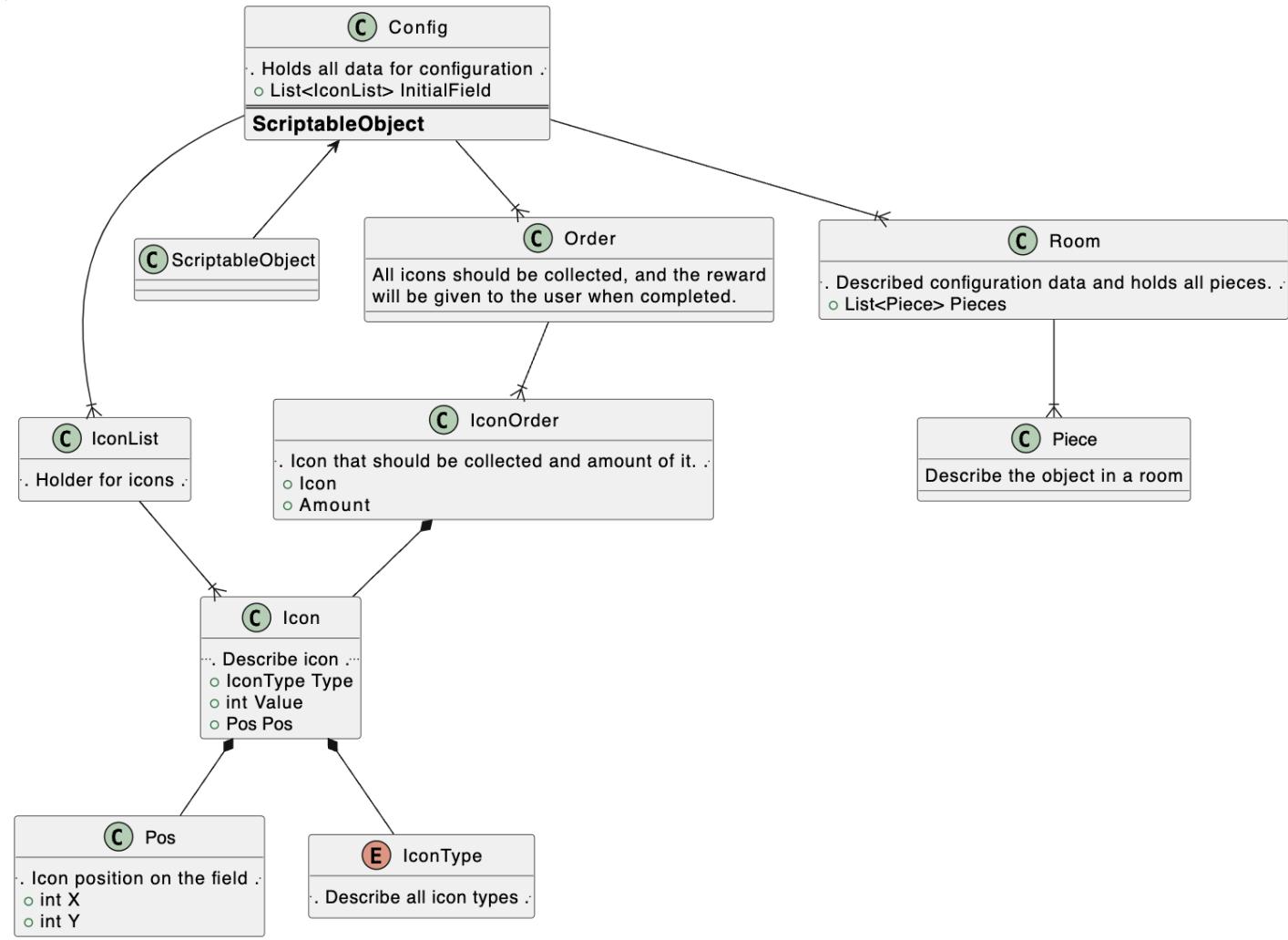
After clicking Apply it will create a new file with the initial field presented on the screen. It will be created in the "SO" folder. That file could be referenced as a new initial field at the Config scriptable object



Technical Design Document

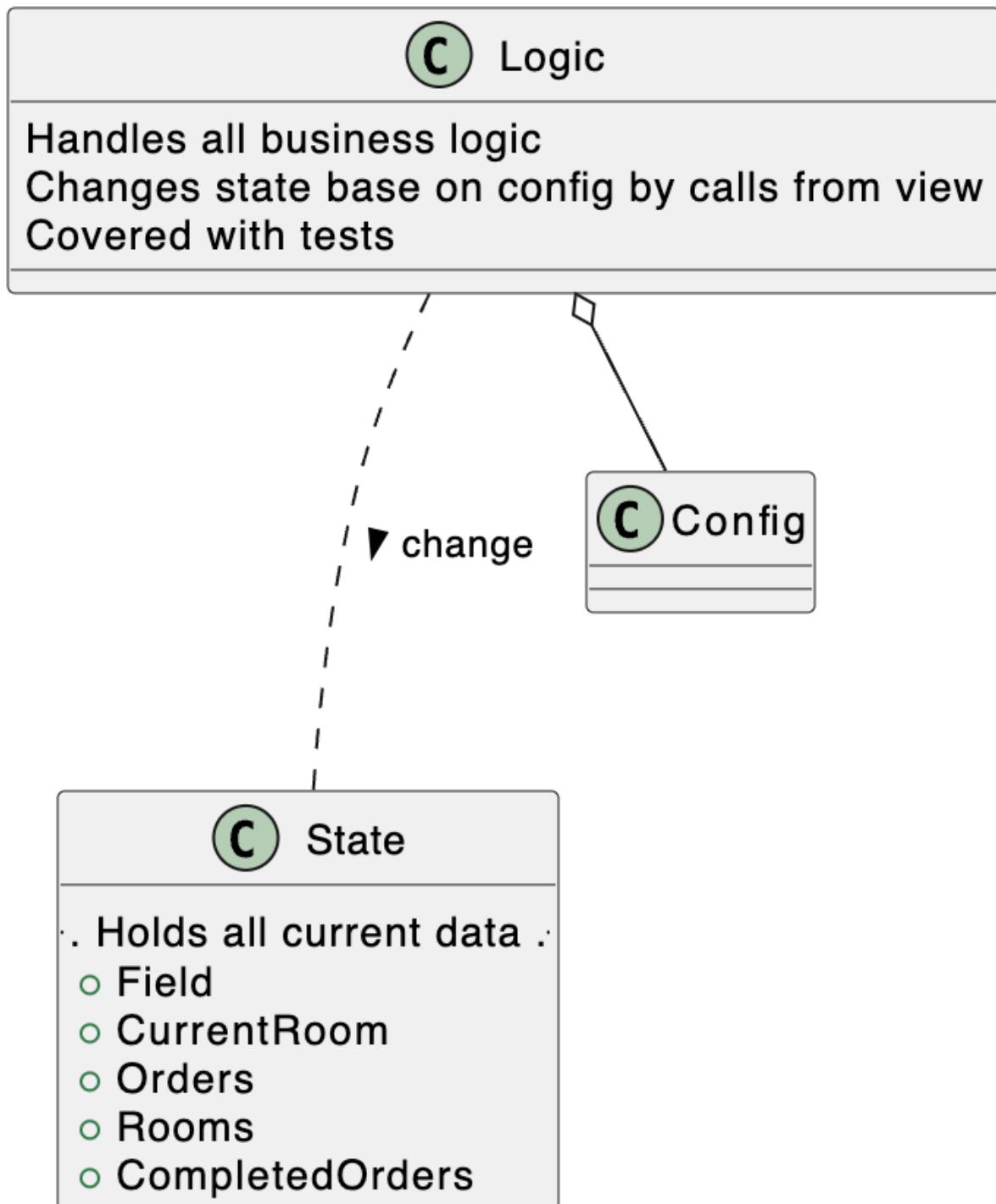
The game is divided into two scenes, Lobby and GameScene. The game works like a synergy of four essences, config, state, logic, and view.

Config - the class where all information about how the game should work, configured data, and sprites are located.

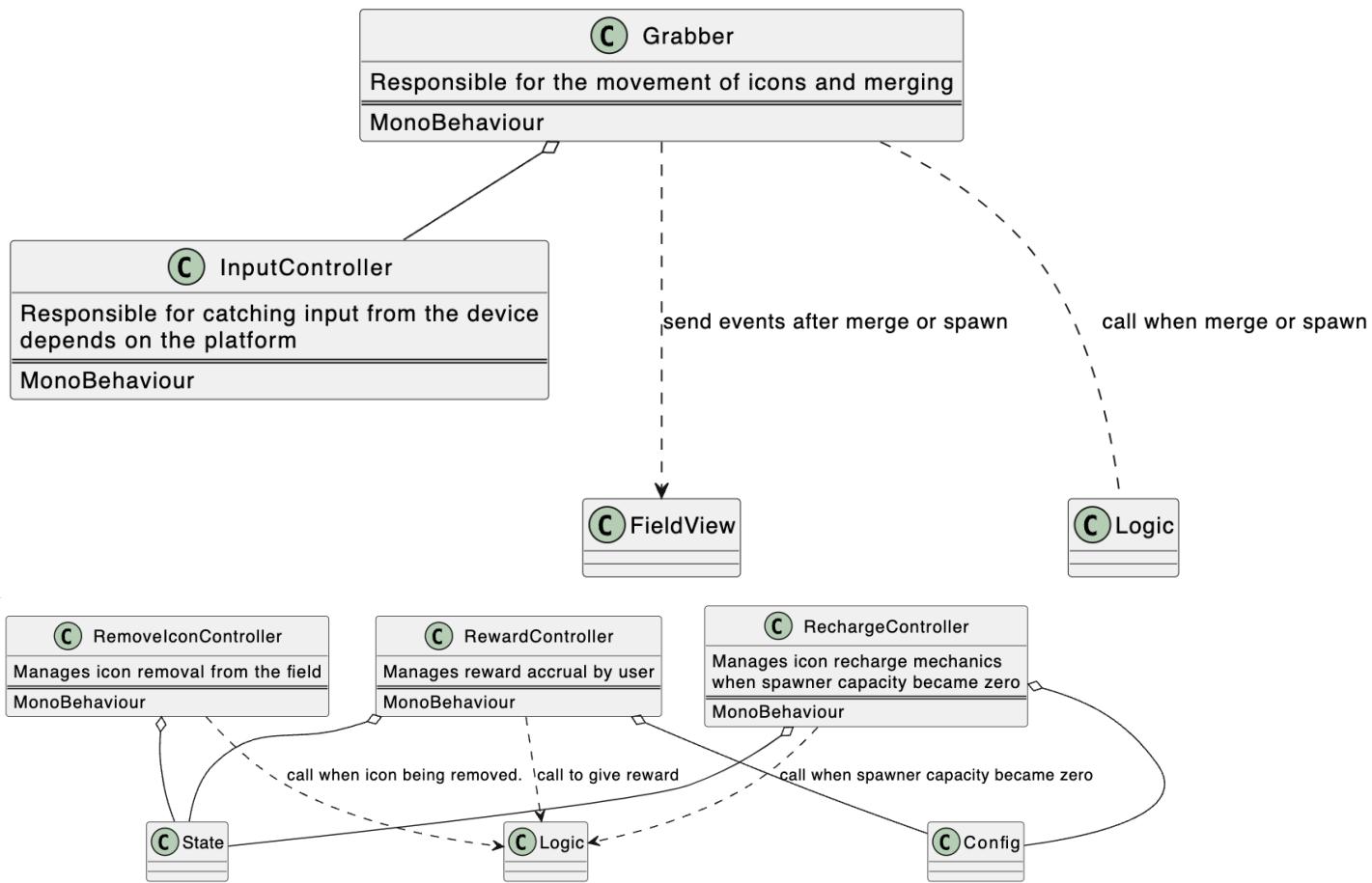


State - the class that holds all current game data. Data about icons, orders, etc...

Logic - the class responsible for making state changes based on user input and configured data.



View - all "MonoBehaviours" that are located in the scenes.



The game's initial field must be configured somehow. To achieve that purpose, the ConfigScene was developed. It allows the designer to set up any icon anywhere in the icon field.

A button randomize was created to speed up the process. The ConfigSceneView script designer seated the list of icons from which random events would happen.

After clicking Apply it will create a new file with the initial field presented on the screen. It will be created in the "SO" folder. That file could be referenced as a new initial field at the Config scriptable object.