

Design Document TicTacToe V1

By Karunam Goyal #138935

Directory Structure of Java Files

```
+FKApplyDesign
  -+src
    -+Players
      - Player.java
      - Machine.java
      - Human.java
    -+State
      - State.java
      - StateManager.java
      - StateUpdater.java
      - Point.java
      - PlayGround.java
      - RunTest.java
    - Compile.sh
    - Run.sh
```

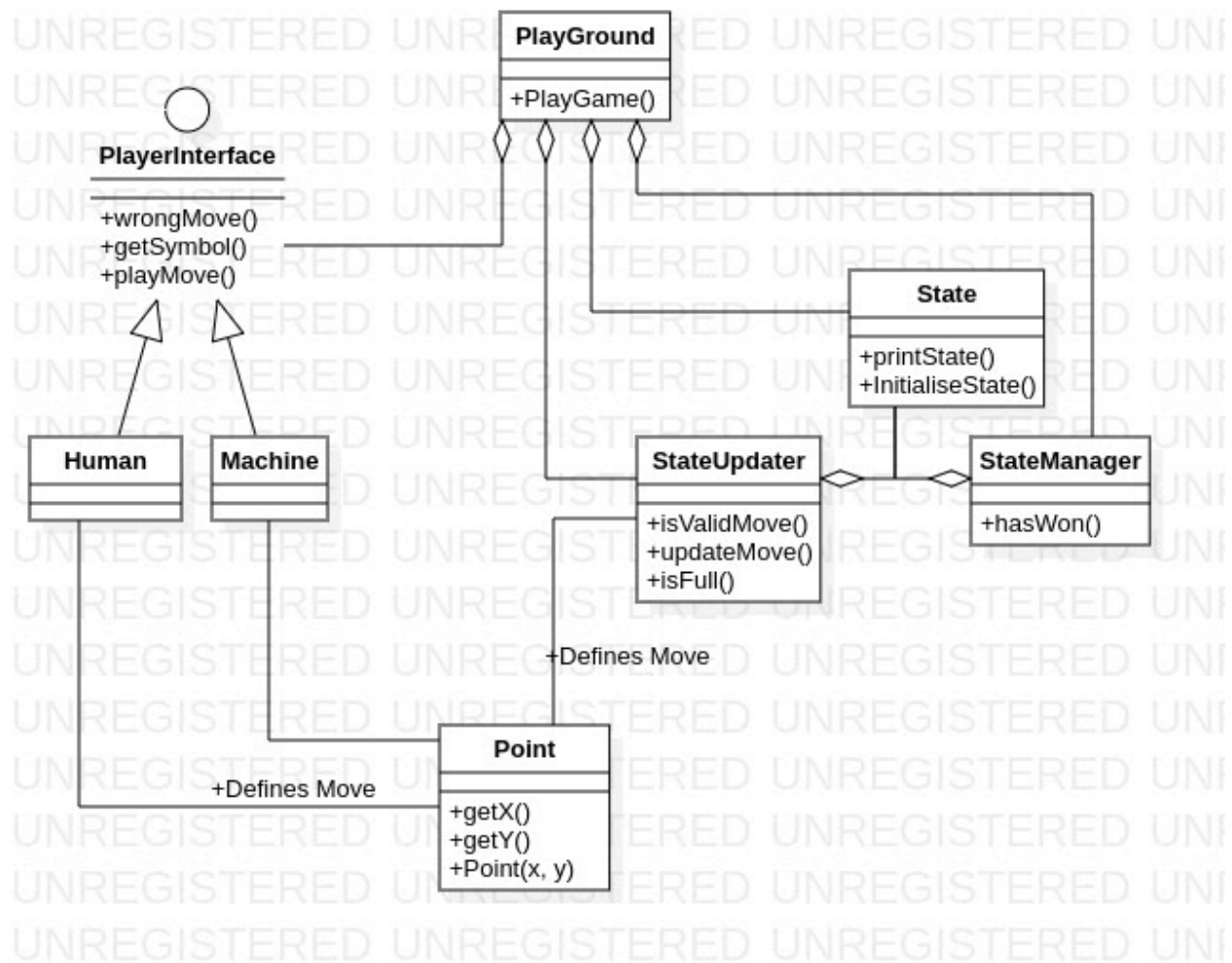
Directory Structure of Class Files

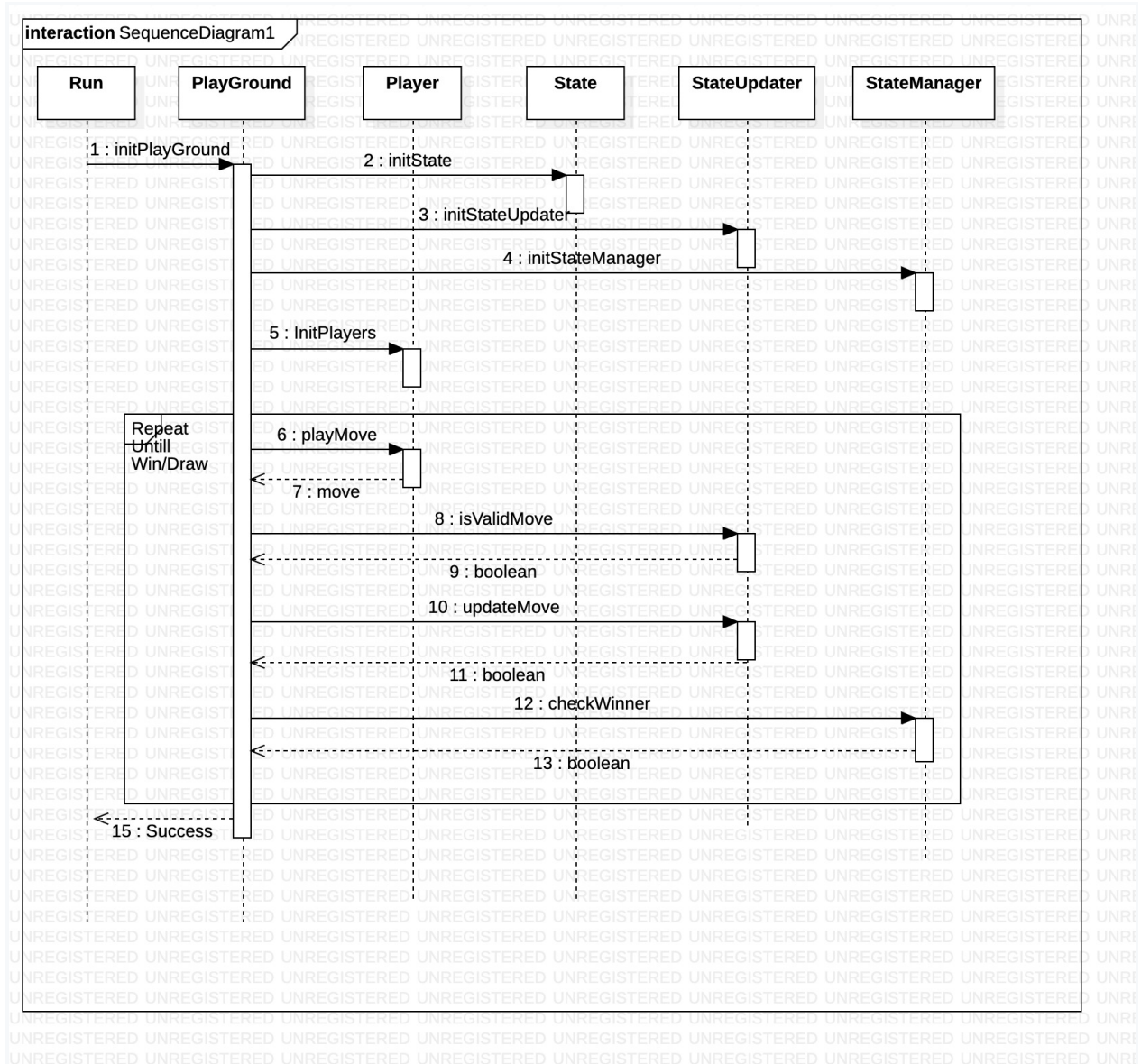
```
+FKApplyDesign
  -+src
    -+Players
      - Player.class
      - Machine.class
      - Human.class
    -+State
      - State.class
      - StateManager.class
      - StateUpdater.class
      - Point.class
    -+Playground
      - PlayGround.class
    - RunTest.class
    - Run.sh
```

Maintainability

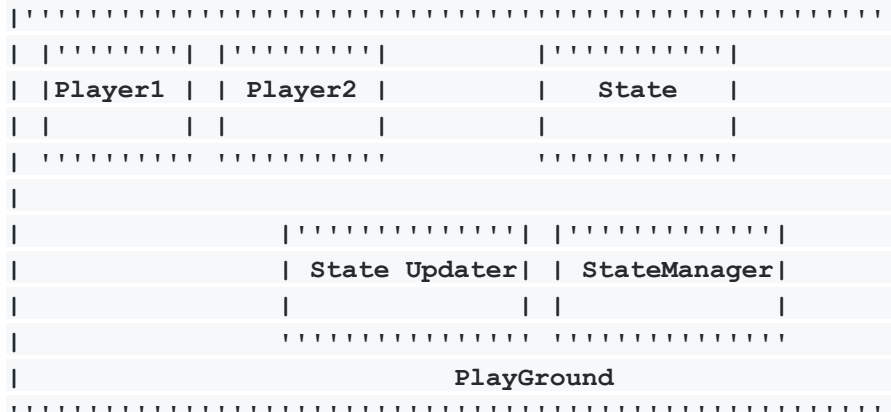
CodeBase is Managed over multiple directories so it is easy for it to manage and maintain the Code for extension and later versions.

Design Plan





Instance of a Game Being Played at a Moment



Playground Class Which Act As A Box Where Game is Played

Player is an Interface act as a Base for Human and Machine Player

Human and **Machine** Class Plays their Moves and they have a symbol they play

State initialises the state and prints the State

StateUpdater as the name Suggests Contains a State Variable and Updates the state and checks its value

StateManager manages the State and Checks if Someone Has Won and returns accordingly

PlayGround Contains the Objects of Players, State, Manager, Updater and the game is played and it act as a box

RunTest Runs The PlayGround

Added Functionality Thinking of Future Requirements (Flexibility)

- Multiple Players
- M X N Row For TicTacToe
- K Check
- Like Tic Tac Toe is (3,3,3)
- Extend it to (M,N,K)
- Change the Data Structure

Why I Think This Design is better

This Design is Good because I've Tried To Do Loose Coupling and use Composition rather than inheriting the things.

This can be further extended to play on multiple systems after adding locking mechanisms or similar Techniques

Other Things which could be was Layered Structure we call the layer above for input and the layers calls other layer for input output and which again passes the data to other layer for processing hence a tightly coupled system hence I created a state which act as a data store only problem is multiple access or race condition when players are playing parallerly but here it is not the case for this structure.