

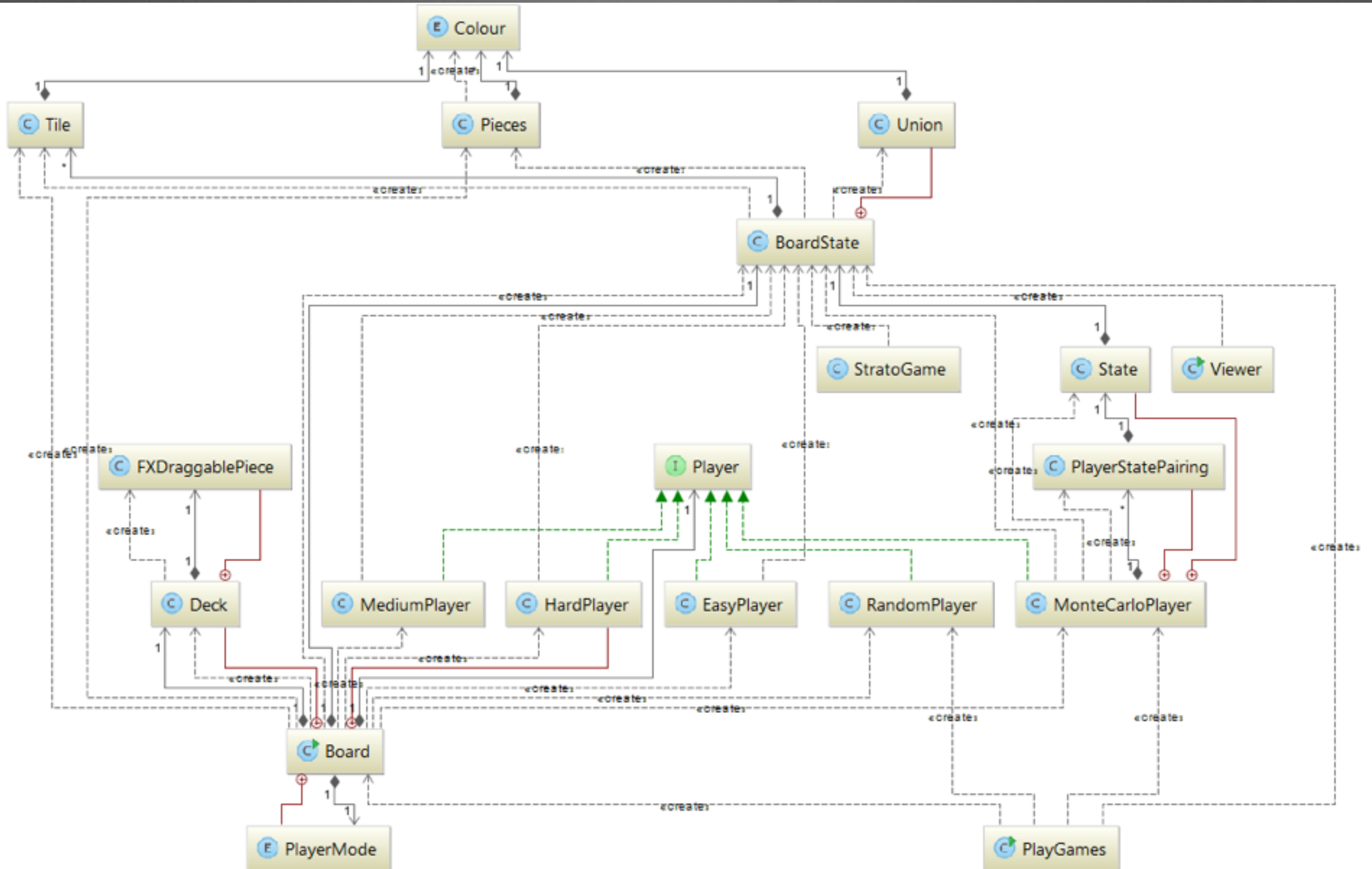
Comp1140 assignment 2

Tom Hamer, Calum Snowdon, Jingyi Lu

Summary


- We will be covering:
 - A demonstration of the game
 - Discussion of the UI and design (Calum)
 - Discussion of BoardState (Calum)
 - Disjoint datasets for getting score
 - Concurrent approaches to get score
 - Discussion of the AI (Tom)
 - Easy, medium and hard (minimax, alphabeta)
 - Neural network
 - Monte Carlo
 - Other features (Jingyi)

UML Diagram




Demonstration

Pieces Remaining: 12



Score: 26

- Human
- Easy AI
- Medium AI
- Hard AI



Pieces Remaining: 13

Score: 24

- Human
- Easy AI
- Medium AI
- Hard AI

Begin Game

UI

- Colour Scheme
- Layout
- Animations
- Text



BoardState

- An abstraction of the board
 - Allowed us to seamlessly implement GetScore, isValid, getMoves, placeTile...
- GetScore disjoint data structure
- Concurrent GetScore
 - too much inefficiency in creating and synchronising threads

AI

- Easy, medium and hard
 - Computational complexity
 - alpha-beta was not enough with our setup
- Neural network
 - attempted to implement a 1 hidden layer nn
 - failed as network was not complex enough to handle the task
 - more hidden layers to increase accuracy
 - convolution layers to decrease computation time
- Monte Carlo
 - failed due to the time taken to find a random move

Other features

- Attempt at save game
 - why it didn't work
- Sound