

Trò chơi ô ăn quan

Học phần :IT3103

Giảng viên : Lê Đức Hậu

Nhóm 8



Thành viên

1. Kha Minh Bao 20210098
2. Le Thi Ngoc Thao 20225673
3. Vo Anh Khoi 20225870
4. Nguyen Tri Dung 20225613
5. Tran Doan Huy 20225859



Thành viên

1. Kha Minh Bảo 20210098
2. Lê Thị Ngọc Thảo 20225673
3. Võ Anh Khôi 20225870
4. Nguyễn Trí Dũng 20225613
5. Trần Doãn Huy 20225859



Mục lục

1. Giới thiệu
2. Thiết kế sơ đồ lớp
3. Phân tích các yếu tố OOP
4. Demo



1. Giới thiệu

- Số người chơi : 2
- Mục đích : chiếm được càng nhiều quan (viên đá, hạt ...) từ đối phương càng tốt





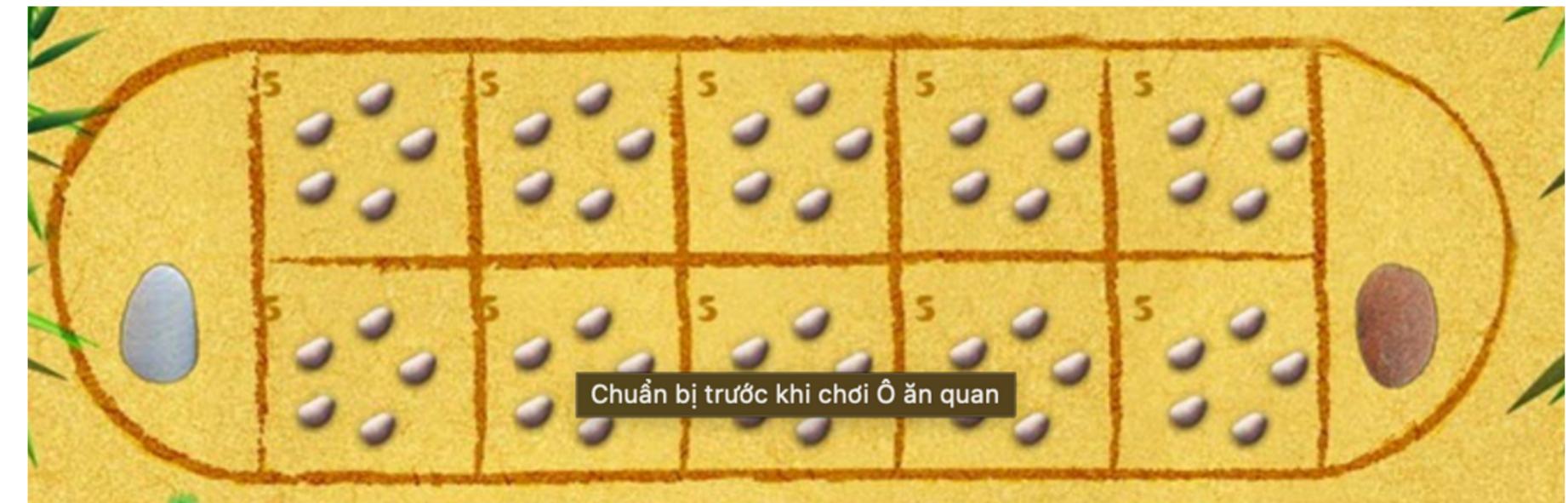
Chuẩn bị

Bàn chơi

- Một hình chữ nhật với hai đầu là hai ô lớn gọi là "ô quan". Phần giữa gồm các ô vuông nhỏ hơn, gọi là "ô dân"

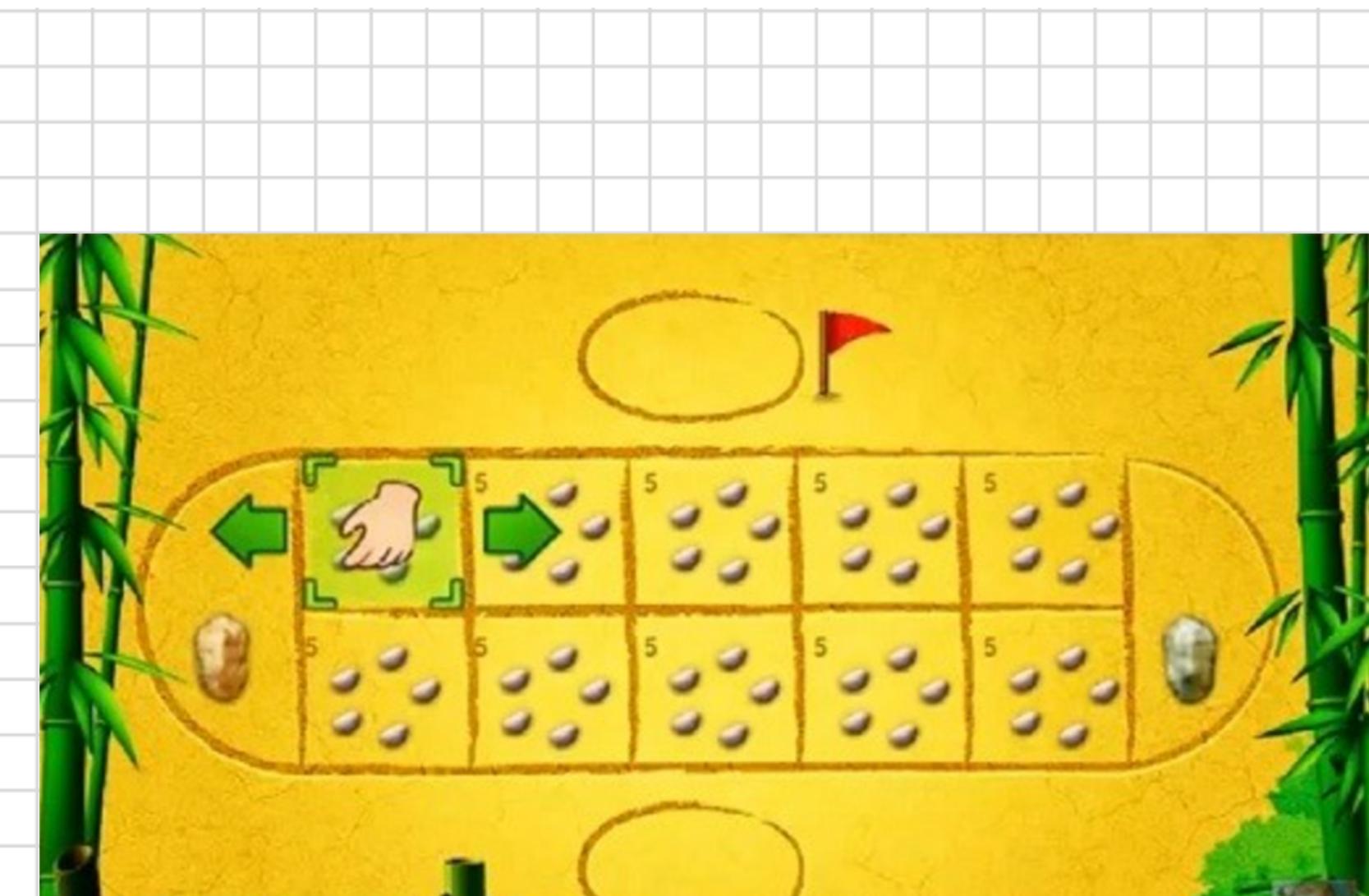
Quân chơi

- Các viên sỏi nhỏ làm dân và hai viên lớn hơn làm quan
- Dân được chia đều vào các ô dân, mỗi ô 5 quân, còn quan được đặt ở hai ô đầu hình chữ nhật



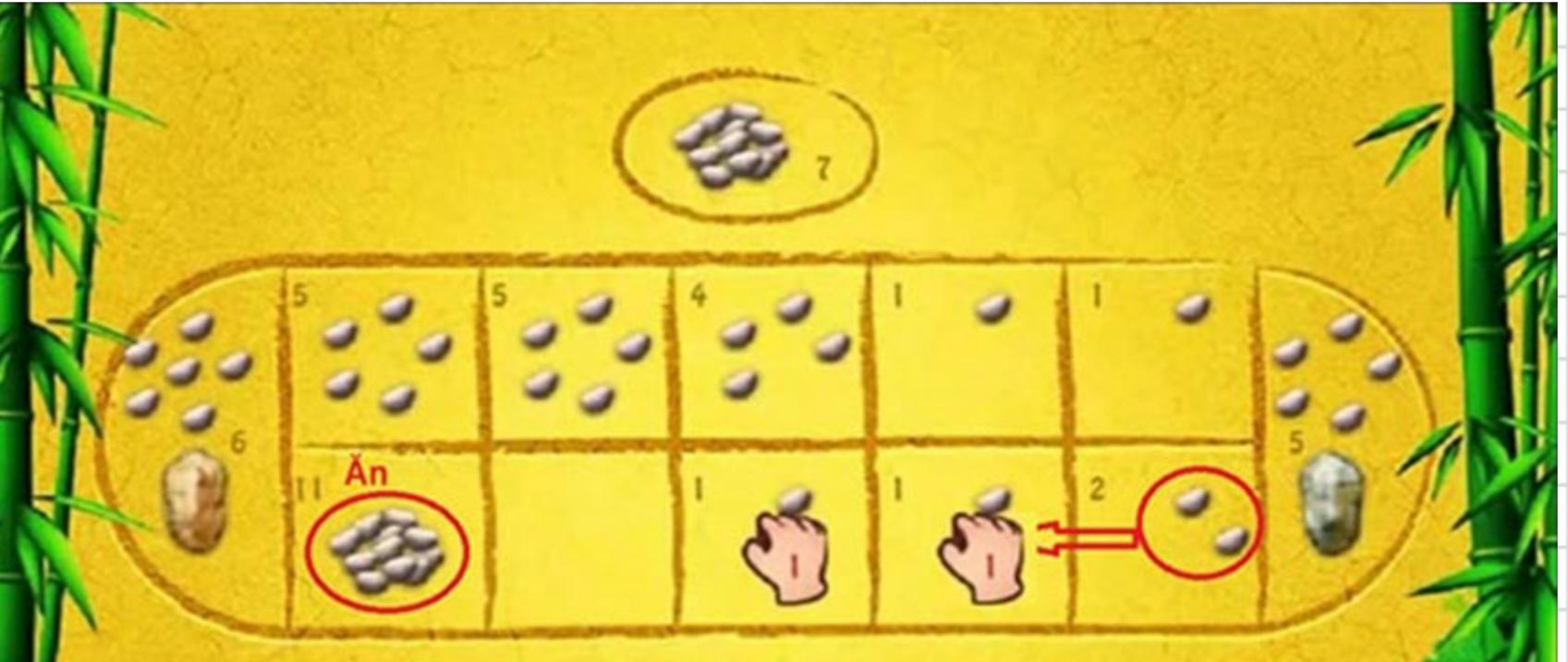
Cách di chuyển

- Người chơi bắt đầu từ một ô con của mình và lấy tất cả các hạt trong ô đó
- Tiến hành "rải" hạt vào các ô con tiếp theo theo chiều đã chọn, một hạt vào mỗi ô cho đến khi hết hạt. -Nếu ô tiếp theo là một ô có chứa quân và không phải ô quan, người chơi sẽ tiếp tục sử dụng số lượng quân ở trong ô đó và tiếp tục di chuyển theo hướng đã được chọn từ trước.
-



Cách di chuyển

- Kết thúc di chuyển, nếu ô tiếp theo trống, người chơi có thể ăn được toàn bộ quân ở ô tiếp sau ô trống.
- Nếu ô sau khi kết thúc là 2 ô trống hoặc ô tiếp theo là ô quan, người chơi sẽ kết thúc lượt của mình



Chú ý luật chơi



Nếu ô quan đã có 1 người làm quan, nếu người chơi có thể ăn được ô quan, chỉ được tính 5 điểm từ vị trí của quan bắt kể có bao nhiêu dân



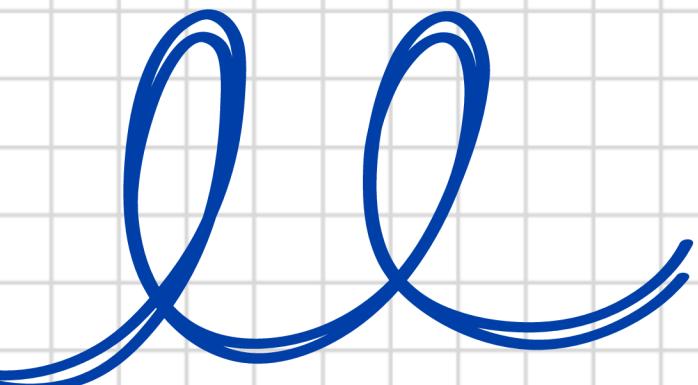
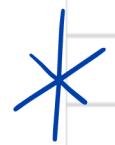
Nếu ô hiện tại chưa có quan thì nếu có dân vào được ô quan, dân đó sẽ trở thành quan và tính 5 điểm nếu ăn được



Nếu đến lượt người chơi mà người chơi phải lấy 5 quân từ điểm của bản thân và dàn đều ra với các vị trí ô, sau đó có thể chọn để đi tiếp



**Trò chơi sẽ kết thúc khi cả 2 ô quan được
ăn và 2 người chơi sẽ được tính điểm để
chọn ra người chiến thắng**



Class Gem (Abstract Class) : mô tả cho các quân

- Bao gồm vị trí và giá trị
- Có 2 class được kế thừa
 - Big Gem (đại diện cho quan) giá trị là 5 điểm
 - Small Gem (đại diện cho dân) giá trị là 1 điểm

Class Cell (abstract class) : Mô tả các ô có trong 1 bàn chơi của 1 ván

- List<gem> : danh sách các cell (trong 1 ván có 12 ô cell)
- int position : vị trí của cell trong board
- boolean isUpperRow: kiểm tra khu vực sân của người chơi số 1 hay 2
- int score: số lượng điểm có trong đó 1 Cell có thể là ô quan hoặc ô dân, trong đó Square được kế thừa thêm 1 interface là Pickable để quyết định xem có thể lấy được quân từ ô đó hay không.

Class Board : mô tả 1 bàn chơi

- 1 bàn chơi bao gồm 12 ô(có 10 ô dân và 2 ô quan)
- Lớp này gồm các thuộc tính liên quan đến trạng thái ban đầu của trò chơi:
 - + Số lượng ô vuông, nửa hình tròn, viên ngọc nhỏ, ngọc lớn
 - + Danh sách các ô (cell) để dễ dàng thao tác với đối tượng
- Trong Constructor của lớp Board: khởi tạo các ô trong ván chơi với index 0 và 6 là các ô quan, và các ô từ 1-5 và 7-11 là các ô dân

Class Player

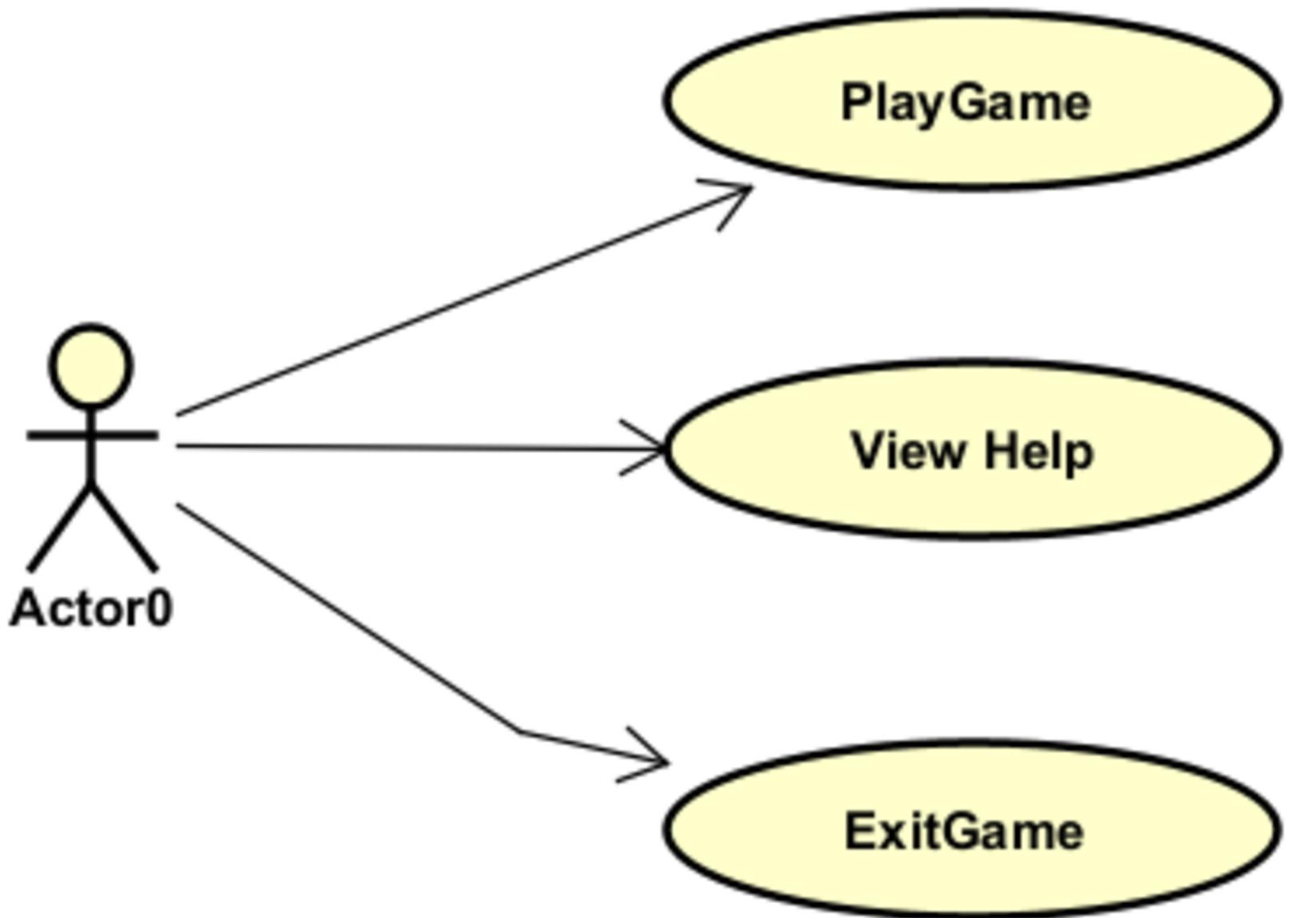
Các phương thức trong lớp này:

- **pickUpGemFrom:** hặt tất cả các viên ngọc từ ô
- **dropGemInto:** rải sỏi vào ô
- **earnGemFrom:** ăn điểm được từ ô đã được chỉ định
- **spreadGem:** rải và kiểm tra điều kiện luật chơi
- **borrowGemsIfNeeded:** kiểm tra và mượn sỏi khi cần thiết
- **areAllCellsEmpty:** kiểm tra xem tất cả các ô chỉ định có trống không
- **addGemsToFirstSquare:** them số lượng sỏi vào square đầu tiên phù hợp

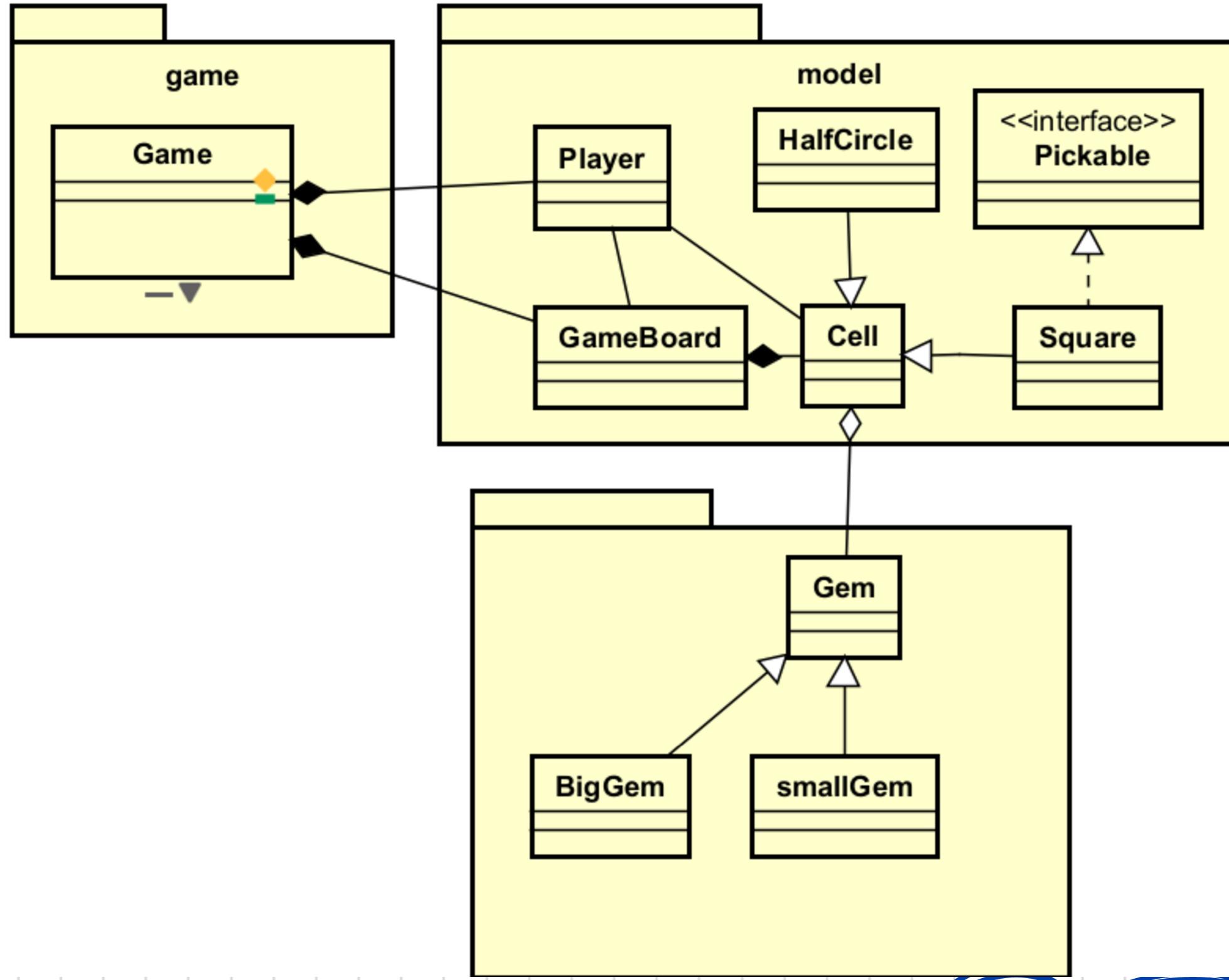
Các class Controller

- **HelpScreenController:** kiểm soát trang helpscreen
- **IntroScreenController:** kiểm soát trang introscreen
- **PlayScreenController** kiểm soát trang Playscreen là trang chính để có thể chơi được trò chơi

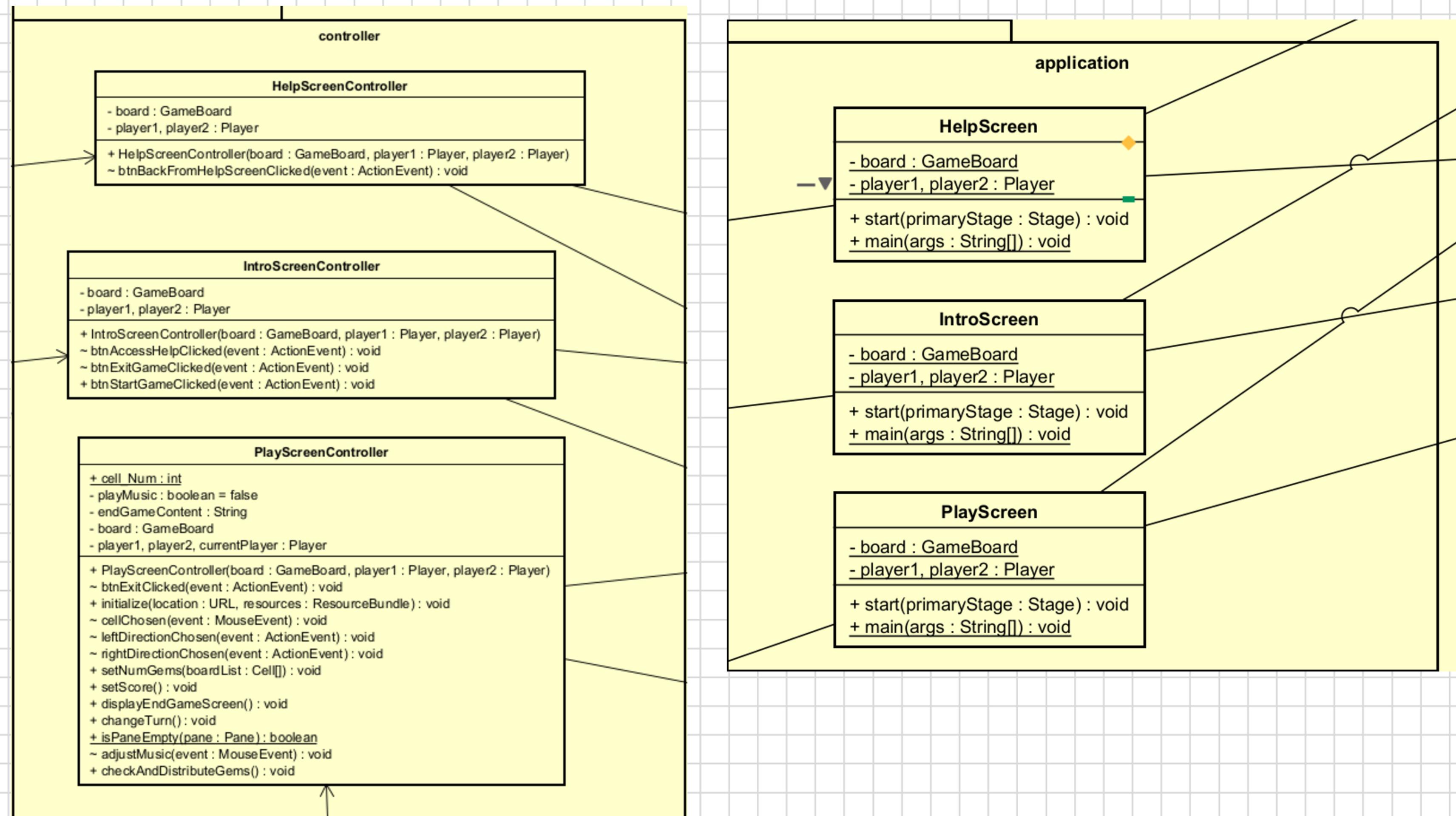
2. Thiết kế sơ đồ lớp



2. Thiết kế sơ đồ lớp



Sơ đồ lớp điều khiển UI



3. Phân tích các yếu tố OOP

- Tính kế thừa: Có thể tạo các lớp con kế thừa từ lớp cha để mô tả các loại lớp khác nhau.

Ví dụ:

- Class BigGem và SmallGem được kế thừa từ class Gem
- Class HalfCircle và Square được kế thừa từ class Cell

- Tính đa hình: Các lớp con có thể định nghĩa lại các phương thức của lớp cha để thực hiện các hành vi khác nhau.

Ví dụ : phương thức addGem sẽ khác nhau đối với class HalfCircle và Square:

Đối với HalfCircle :

```
public void addGem(Gem gem) {  
    getGems().add(gem);  
}
```

Đối với Square:

```
public void addGem(Gem gem) {  
    gems.add(gem);  
}
```

- Tính đóng gói: Ẩn các thuộc tính và phương thức bên trong lớp và chỉ cung cấp các phương thức truy cập công khai để bảo vệ dữ liệu.

ví dụ : bảo vệ private các giá trị value và position

4. Demo chương trình