BÁO CÁO ĐỒ ÁN LẬP TRÌNH C#

XẾP HÌNH CỔ ĐIỂN -Tetris

Giảng viên hướng dẫn:

TS.Bùi Tiến Lên

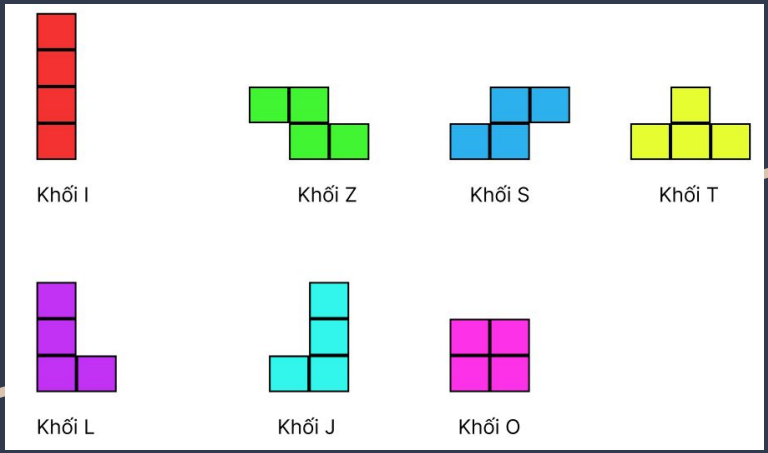
Thành viên nhóm 2:

Lê Thị Mỹ Hương	3122411077
Đỗ Minh Quân	3122411166
Trần Bùi Ty Ty	3122411241

NỘI DUNG

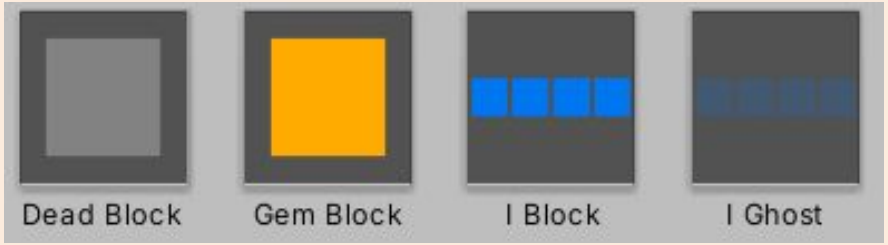
Giới thiệu đề tài

~Trò chơi xếp hình cổ điển - Tetris~



- Tetris là trò chơi xếp hình kinh điển, nơi người chơi điều khiển các khối Tetriminos để lấp đầy hàng ngang và ghi điểm.
- Lối chơi đơn giản nhưng thách thức, giúp rèn luyện tư duy logic, phản xạ nhanh và kỹ năng xử lý tình huống thông qua độ khó của game
- Gồm 7 hình dạng đặc trưng: I, O, T, L, J, S, Z, mỗi khối mang một chiến thuật riêng và góp phần tạo nên sự cuốn hút.

THUẬT NGỮ



Khối (Block): Các mảnh ghép với 7 dạng hình học chính (Tetriminos).

GemBlock: Khối đặc biệt trong chế độ “Màn chơi” (Khối màu cam).

GhostBlock: Hiển thị vị trí rơi của khối.

DeadBlock: Khối không thể di chuyển.

Chế độ chơi: Chơi Theo Màn và Chơi Vô Hạn.

Xuống Nhanh/Xuống Lập Tức: Điều khiển tốc độ rơi của khối.

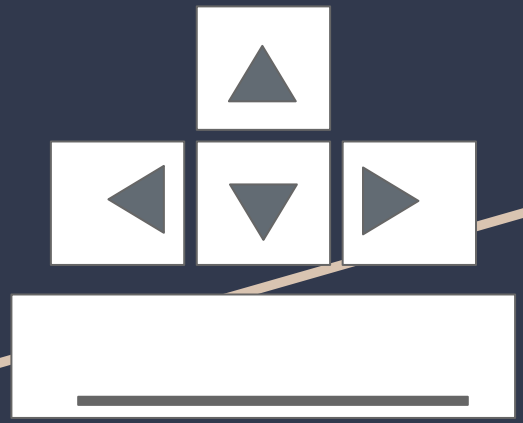
Xóa Dòng: Xóa hàng khi lấp đầy.

Ghi Điểm: Điểm được ghi nhận dựa trên số dòng xóa được.

Xoay Khối/Di chuyển Khối/ Khối Kế Tiếp: di chuyển khối/xem trước các khối.

Phân tích yêu cầu

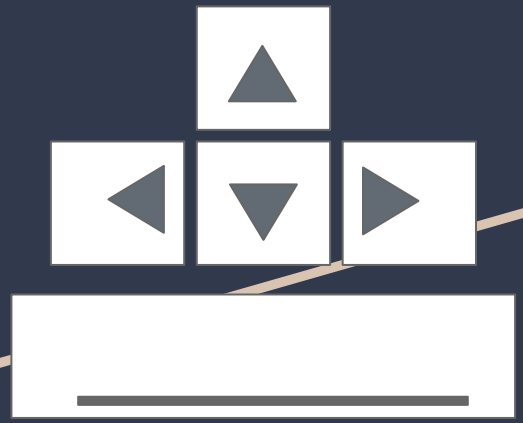
~Mô tả tổng quan trò chơi~



- Bàn chơi: Lưới 10x20, nơi các khối Tetrimino rơi từ đỉnh xuống đáy.
- Điều khiển Tetrimino:
  - Xoay khối: Xoay 90° quanh điểm xoay.
  - Di chuyển khối: Trái/phải để tránh va chạm.
  - Hạ nhanh:
    - Soft Drop: Tăng tốc độ rơi.
    - Hard Drop: Đưa khối xuống đáy ngay lập tức.
- Đóng băng khối: Khối cố định khi chạm đáy hoặc va khối khác.

Phân tích yêu cầu

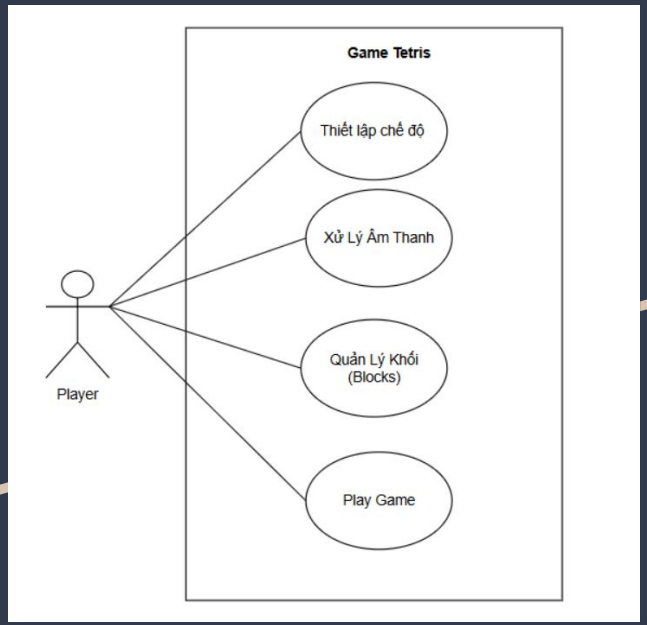
~Mô tả tổng quan trò chơi~



- Mục tiêu:
  - Vô hạn: Ghi điểm cao nhất có thể
  - Màn chơi: Xóa GemBlocks, vượt 10 màn để hoàn thành trò chơi.

Phân tích yêu cầu

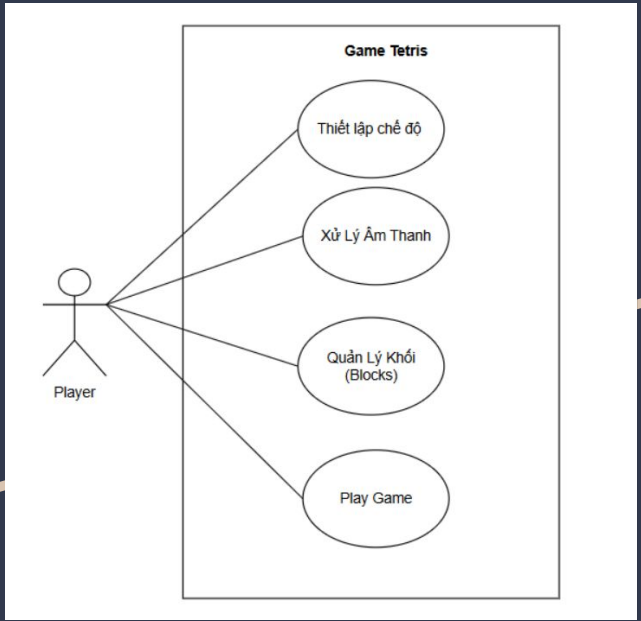
~Mô tả hệ thống~



STT	Tên Chức Năng	Mô Tả
UC01	Thiết lập chế độ chơi	Cho phép người chơi thiết lập chế độ chơi bao gồm chế độ "vô hạn" hoặc "theo màn" và xem hướng dẫn.
UC02	Xử lý âm thanh	Hệ thống tự động phát âm thanh nền khi trò chơi bắt đầu, dừng âm thanh khi kết thúc và chuyển âm thanh khi thay đổi màn chơi.

Phân tích yêu cầu

~Mô tả hệ thống~

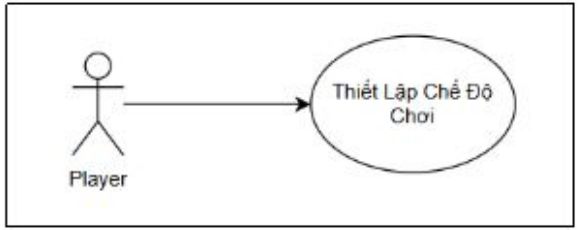


UC03	Quản lý khối (Blocks)	Người chơi thao tác để điều khiển khối rơi bằng các phím: tăng tốc (xuống), rơi thẳng (cách), xoay 90° (lên), và di chuyển ngang (trái/phải).
UC04	Play Game	Quản lý toàn bộ quá trình chơi game: bắt đầu, tạm dừng, xử lý khối rơi, kiểm tra trạng thái thắng/thua, và lựa chọn chơi lại hoặc thoát.

Phân tích yêu cầu

~Mô tả hệ thống~

-UC01-



Use Case Number:	**UC01**	
Use Case Name:	Thiết lập chế độ chơi	
Actor (s):	*Player*	
Maturity:	Focused	
Summary:	*Thiết lập chế độ chơi hoặc hiển thị hướng dẫn trò chơi theo nhu cầu, sở thích của Player. Player có thể chọn chế chơi "vô hạn", hoặc "theo màn"*	
Basic Course of Events:	**Actor Action**	**System Response**
	1. Actor truy cập vào game.	
		2. Hệ thống hiển thị trang chủ game.
	3. Actor chọn chế độ. **A1**	
		4. Hệ thống thiết lập chế độ chơi.

Phân tích yêu cầu

~Mô tả hệ thống~

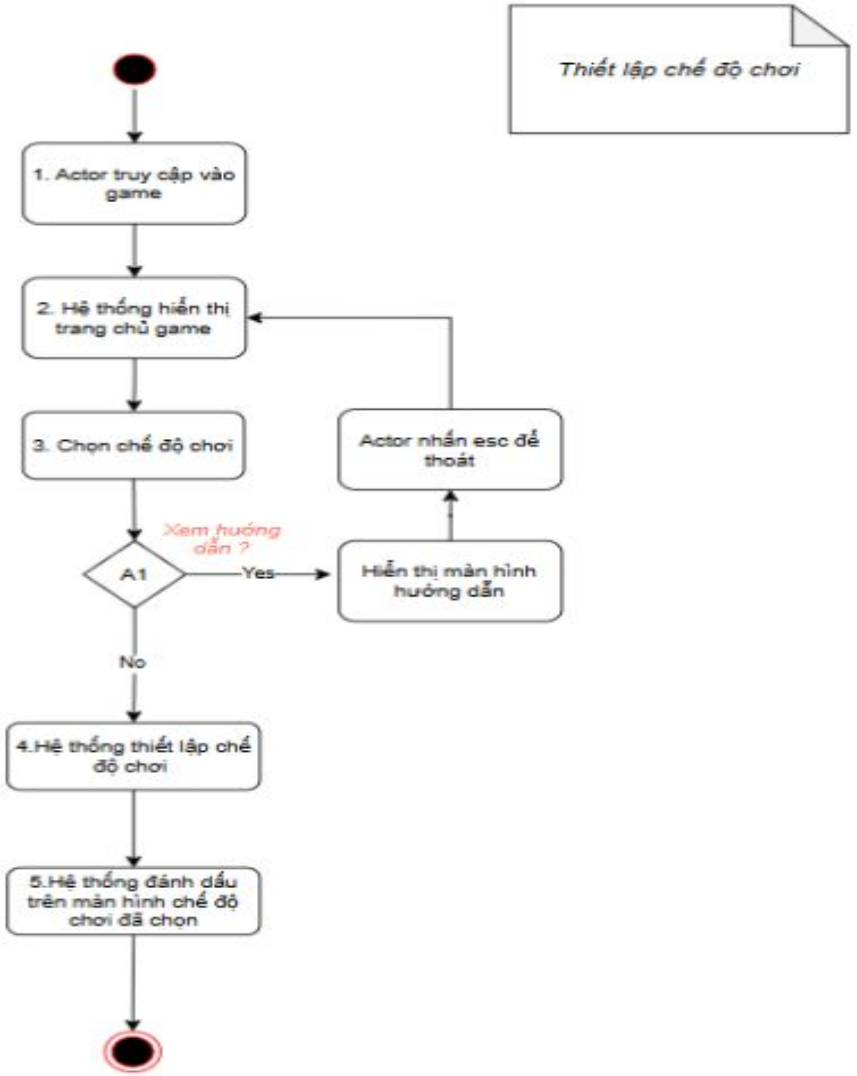
-UC01-

		5. Hệ thống đánh dấu trên màn hình chế độ chơi đã được chọn. **Use Case kết thúc tại đây**
Alternative Paths:	**A1. Xem Hướng Dẫn**	
	Actor Action	**System Response**
	1. Actor chọn "Hướng Dẫn".	
		2. Hệ thống hiển thị màn hình hướng dẫn.
	3. Actor nhấn Esc để thoát. **Quay lại bước 2 UC1 trong Basic Course of Events**	
Exception Paths:	None	
Extension Points:	None	
Triggers:	None	
Assumptions:	None	
Preconditions:	None	
Post Conditions:	None	
Reference: Business Rules	None	
Author(s):	Nhóm 2	
Date:	16-12-2024	

Phân tích yêu cầu

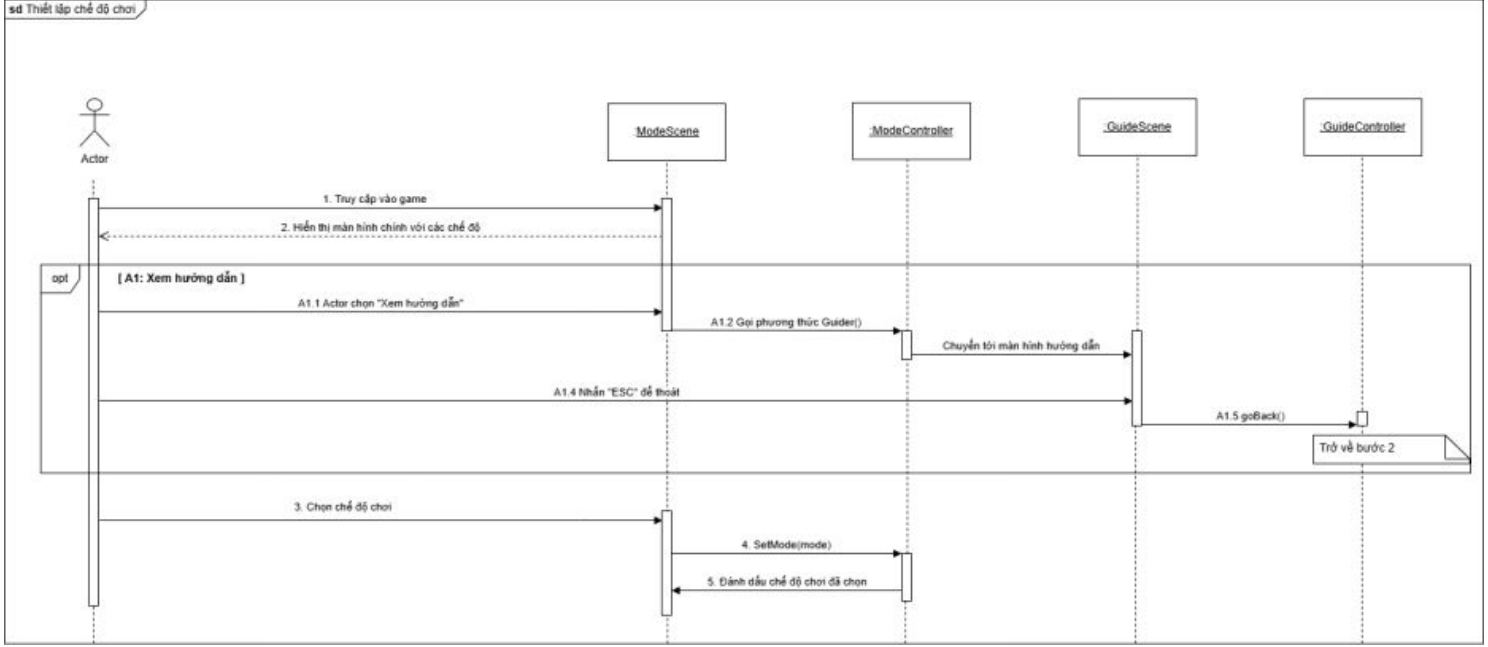
Activity diagram

UC01



Giao diện Thiết Lập Chế Độ

Phân tích yêu cầu

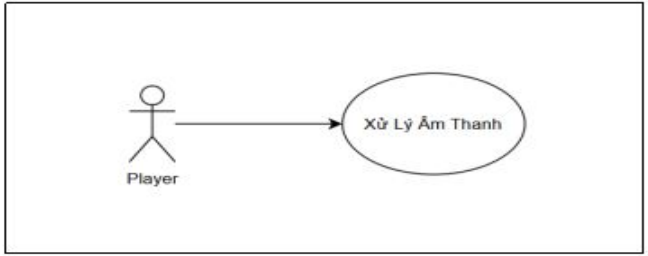


-UC01-

Phân tích yêu cầu

~Mô tả hệ thống~

-UC02-



Use Case Number:	UC02	
Use Case Name:	Xử Lý Âm Thanh	
Actor (s):	*Player*	
Maturity:	Focused	
Summary:	*Player thao tác để phát âm thanh nền và các trạng thái tắt/mở âm thanh của hệ thống.*	
Basic Course of Events:	**Actor Action**	**System Response**
	1. Actor truy cập vào game	
		2. Hệ thống tự động phát âm thanh nền khi trò chơi bắt đầu.
	3. Player thao tác để đặt khối hoặc để khối tự rơi. **A1**	
		4. Hệ thống kiểm tra điều kiện kết thúc game. **A2**

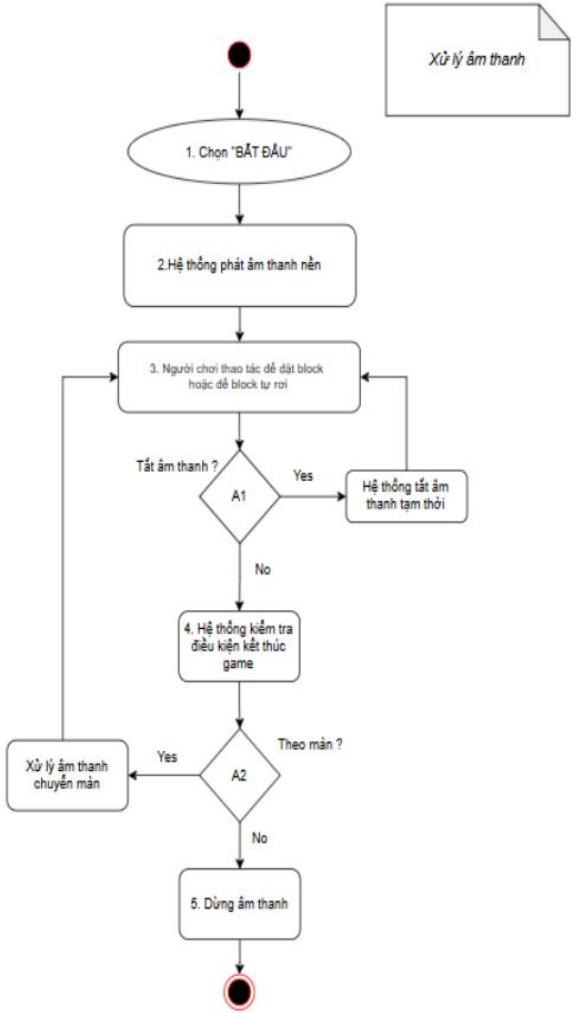
Phân tích yêu cầu

~Mô tả hệ thống~

-UC01-

		5. Hệ thống dừng âm thanh. **Use Case kết thúc tại đây**
Alternative Paths:	**A1.** Hệ thống kiểm tra màn chơi hiện tại	
	A2. Xử lý âm thanh chuyển màn.	
Exception Paths:	**None**	
Extension Points:	**None**	
Triggers:	Khi Actor bắt đầu trò chơi	
Assumptions:	**None**	
Preconditions:	**None**	
Post Conditions:	**None**	
Reference: Business Rules	**None**	
Author(s):	Nhóm 2	
Date:	16-12-2024	
Activity Diagram: Xử Lý Âm Thanh		

Activity diagram UC02

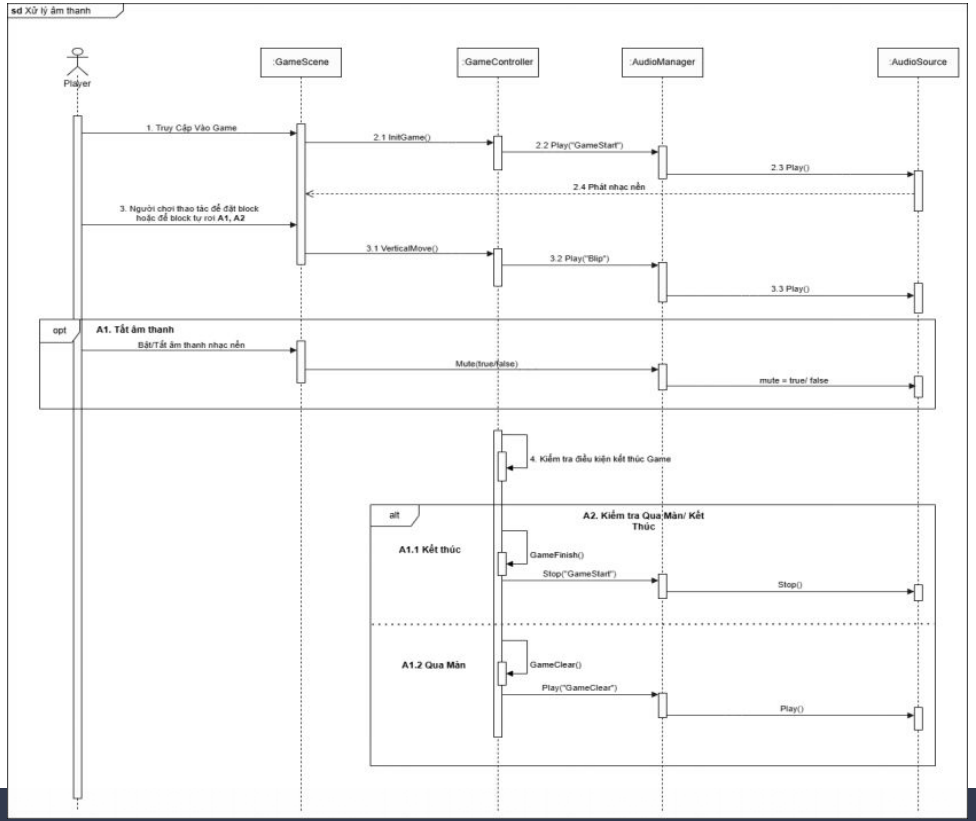


Giao diện Xử Lý Âm Thanh

Phân tích yêu cầu

~Mô tả hệ thống~

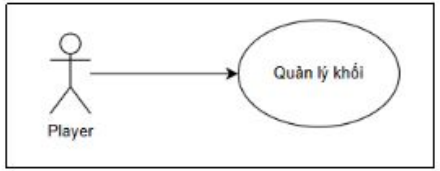
-UC02-



Phân tích yêu cầu

~Mô tả hệ thống~

-UC03-



Use Case Number:	UC3	
Use Case Name:	Quản lý khối	
Actor (s):	*Player*	
Maturity:	Focused	
Summary:	*Player thao tác để điều khiển di chuyển khối đang rơi hiện tại*	
Basic Course of Events:	**Actor Action**	**System Response**
	1. Actor bắt đầu chơi game	
		2. Hệ thống hiển thị giao diện chơi game
		3. Khối tự rơi xuống theo thời gian
	4. Actor chọn thao tác **A1, A2, A3, A4.**	
		4. Khối xuất hiện ở vị trí cuối. **Use Case kết thúc**

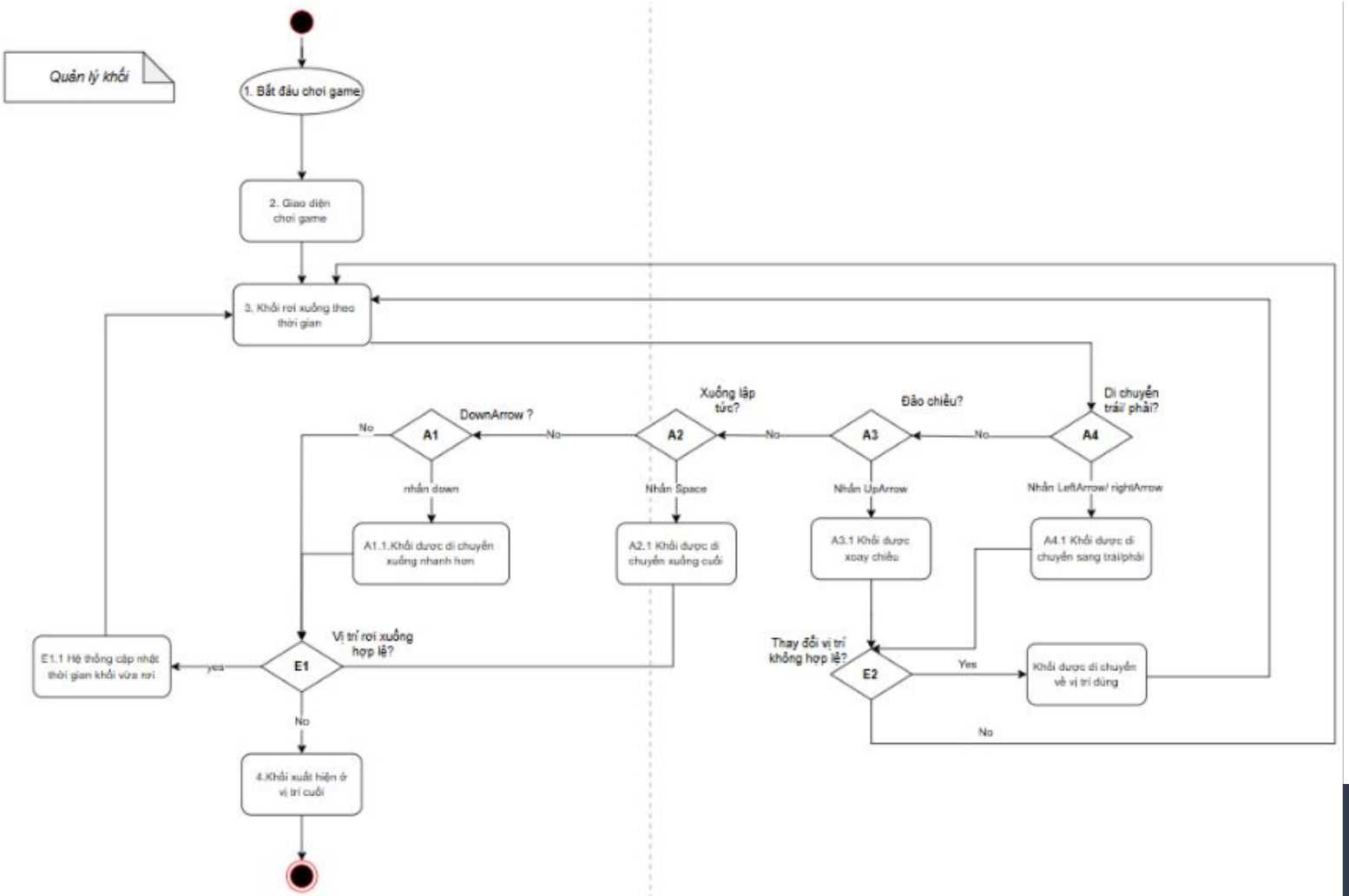
Phân tích yêu cầu

~Mô tả hệ thống~

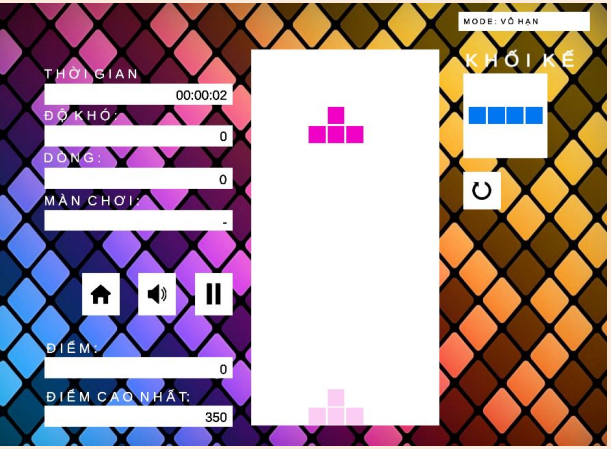
-UC03-

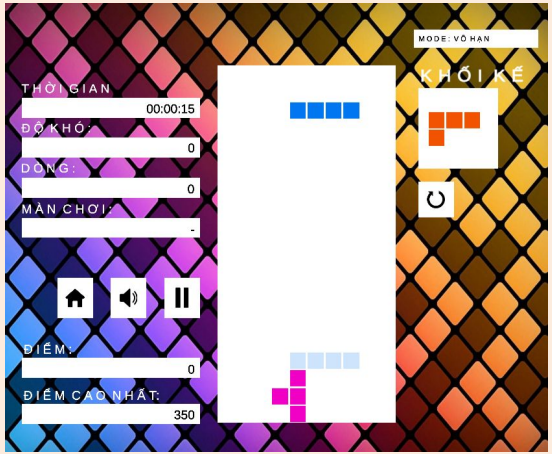
Alternative Paths:	**A1.**Nút downArrow được nhấn thì khối rơi xuống nhanh hơn **E1. Quay về bước 6 UC03**
	A2. Nút Space được nhấn thì khối rơi xuống vị trí cuối **E1 Quay lại bước 3 UC03**
	A3 Nút up Arrow được nhấn thì khối đảo chiều 90 độ **E2 Quay lại bước 3 UC03**
	A4. Nút LeftArrow/ rightArrow được nhấn thì khối được di chuyển sang trái/ phải **E2 Quay lại bước 3 UC03**
Exception Paths:	**E1** Nếu vị trí rơi xuống hợp lệ thì hệ thống cập nhật thời gian vừa rơi, **Quay lại bước 3 UC03**
	E2 Nếu vị trí thay đổi không hợp lệ thì hệ thống hiển thị khối ở vị trí đúng **Quay lại bước 3 UC03.**
Extension Points:	**None**
Triggers:	Khi Actor bắt đầu trò chơi
Assumptions:	**None**
Preconditions:	**None**
Post Conditions:	**None**
Reference: Business Rules	**None**
Author(s):	Nhóm 2
Date:	16-12-2024
Activity Diagram: Quản lý khối	

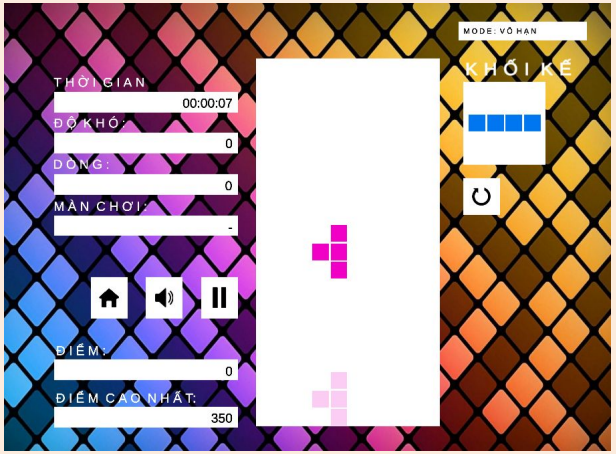
Activity diagram UC03



Giao diện Xử Lý Khối



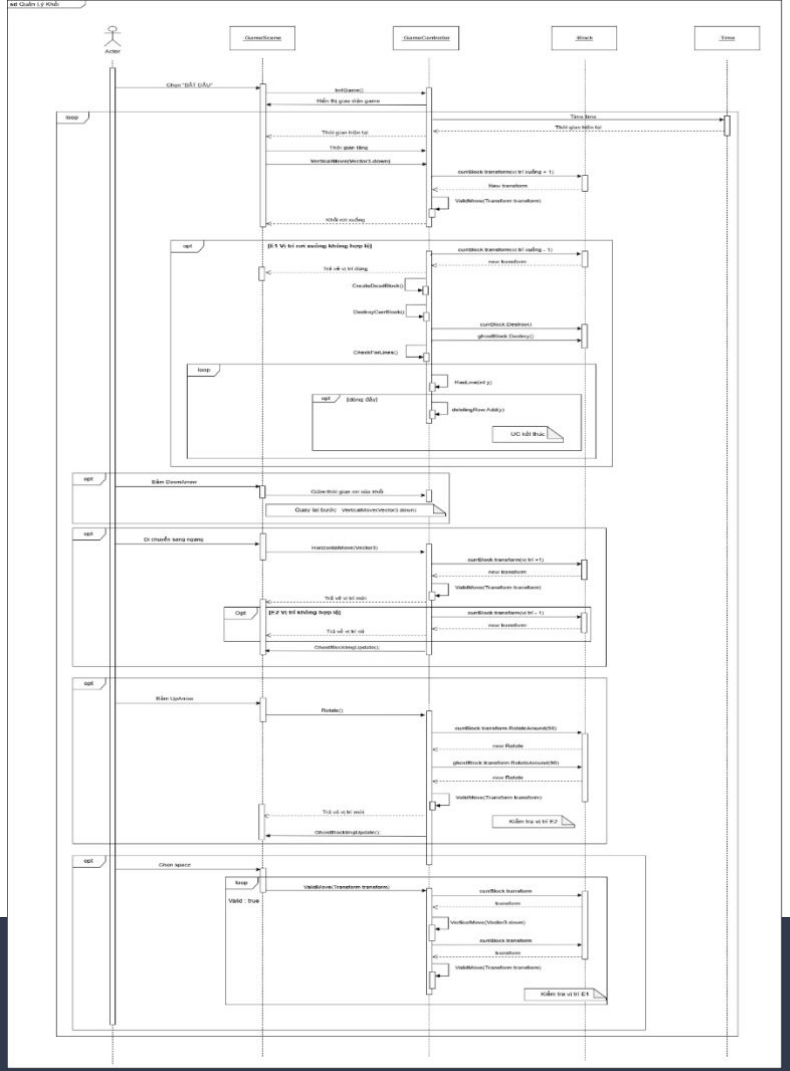




Phân tích yêu cầu

~Mô tả hệ thống~

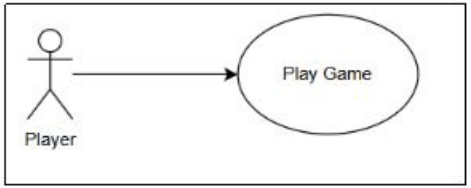
-UC03-



Phân tích yêu cầu

~Mô tả hệ thống~

-UC04-



Use Case Number:	UC04	
Use Case Name:	Play Game	
Actor (s):	*Player*	
Maturity:	Focused	
Summary:	*Actor thực hiện các thao tác bắt đầu trò chơi, chơi game, xử lý các khối và kết thúc trò chơi, với các lựa chọn chơi lại hoặc thoát về màn hình chính.*	
	Actor Action	**System Response**
	1. Actor bắt đầu chơi game	
		2. Hệ thống Xử Lý Âm Thanh **A1**
		3. Hệ thống hiển thị giao diện chơi "Theo màn"
		4. Hệ thống bắt đầu tạo khối chơi (Khối chính và Khối ảo)
	5. Actor thao tác chơi game **A2, A3, A4**	
		6. Hệ thống thao tác Xử Lý Khối **A5**
		7. Hệ thống xóa hàng và cập nhật điểm **A6**

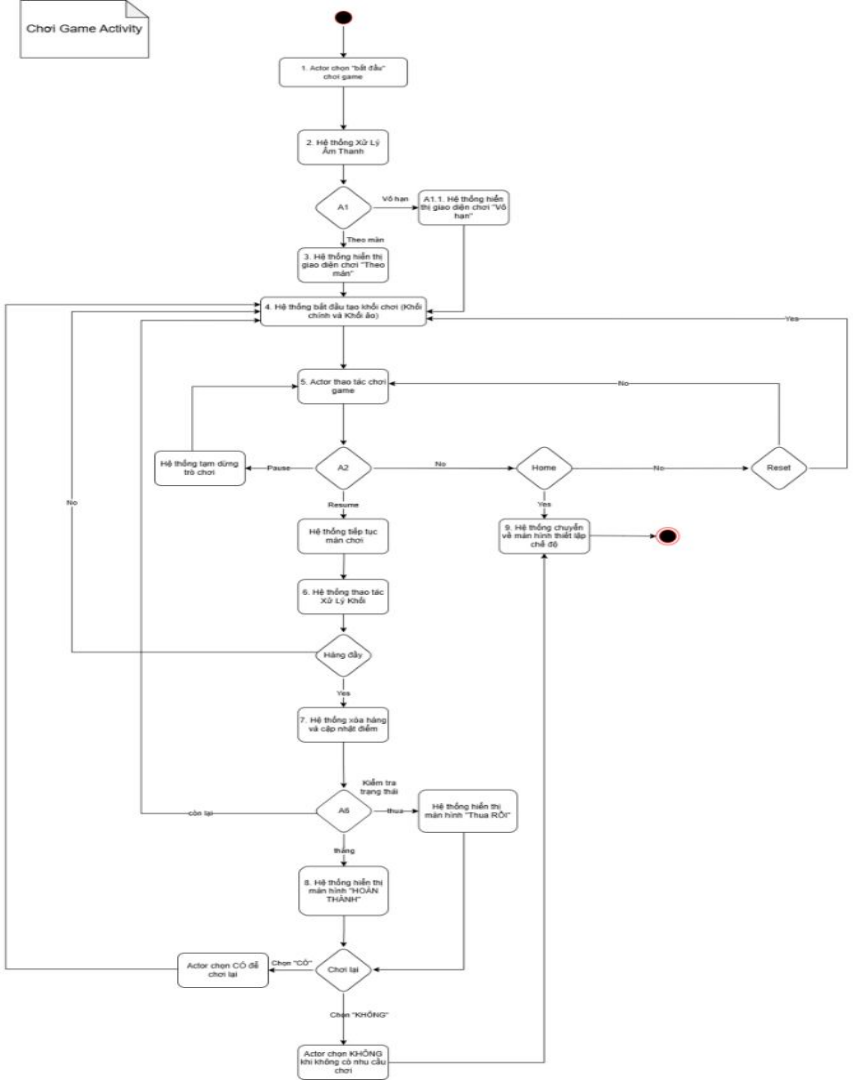
Phân tích yêu cầu

~Mô tả hệ thống~

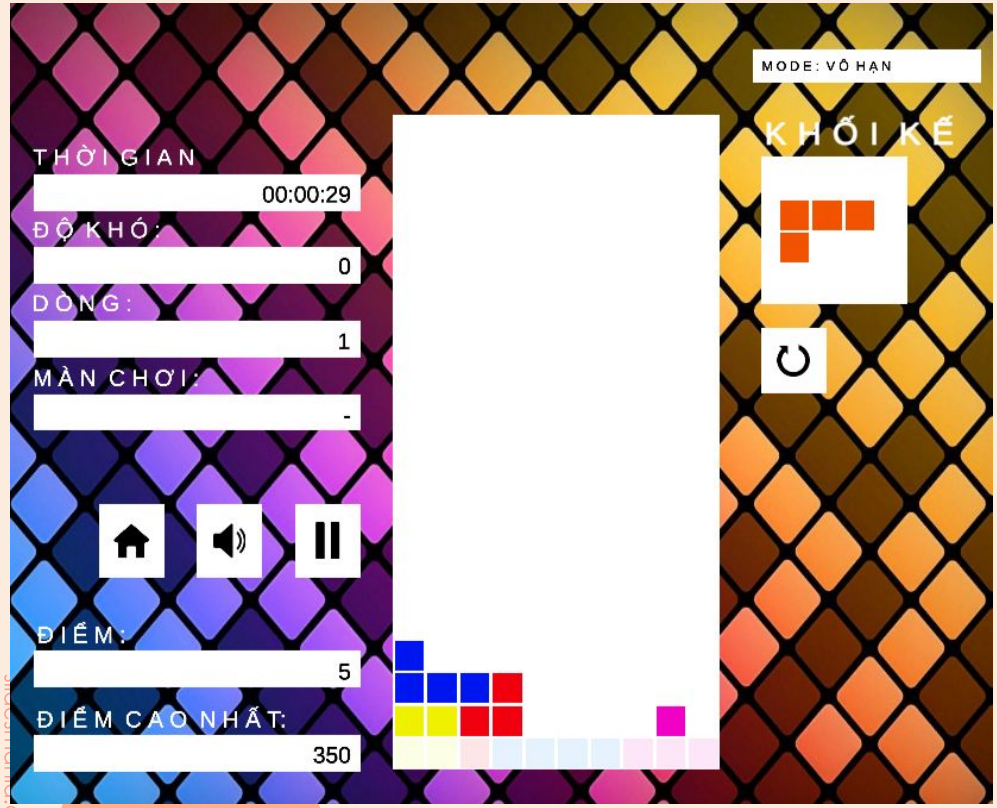
-UC04-

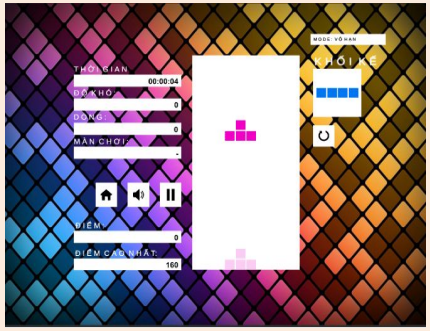
		8. Hệ thống hiển thị màn hình "HOÀN THÀNH" **A7**
	9. Actor chọn "Không" khi không có nhu cầu chơi lại **A7**	
		9. Hệ thống chuyển về màn hình thiết lập chế độ **Use Case kết thúc tại đây**
Alternative Paths:	**A1.** Hệ thống hiển thị giao diện chơi "Vô hạn". **Quay về bước 4 UC04**	
	A2 Pause/ Resume **A2.1** Hệ thống tiếp tục màn chơi **Quay về bước 6 UC04** **A2.2** Hệ thống tạm dừng trò chơi **Quay về bước 5 UC04**	
	A3 Actor chọn home **Quay lại bước 9 UC04**	
	A4. Actor chọn reset **Quay lại bước 4 UC04** nếu không **Quay lại bước 5 UC04**	
	A5. kiểm tra hàng đầy Nếu hàng đầy **Quay lại bước 7 UC04** nếu không **Quay lại bước 4 UC04**	
	A6. Kiểm tra tình trạng **A6.1** Nếu thua Hệ thống hiển thị màn hình "thua rồi" **A7** **A6.2** Nếu thắng hệ thống hiển thị màn hình "Hoàn thành" **A7** **A6.3** Nếu không **Quay lại bước 4 UC04**	
	A7. Người chơi chọn "Có" để chơi lại. **Quay lại bước 4 UC04**	
Exception Paths:	**E1** Nếu vị trí rơi xuống hợp lệ thì hệ thống cập nhật thời gian vừa rơi, **Quay lại bước 3 UC03**	
	E2 Nếu vị trí thay đổi không hợp lệ thì hệ thống hiển thị khối ở vị trí đúng **Quay lại bước 3 UC03.**	
Extension	**None**	

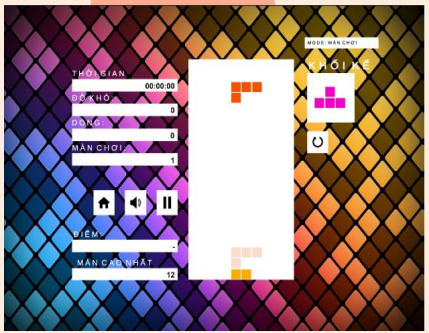
Activity diagram UC04

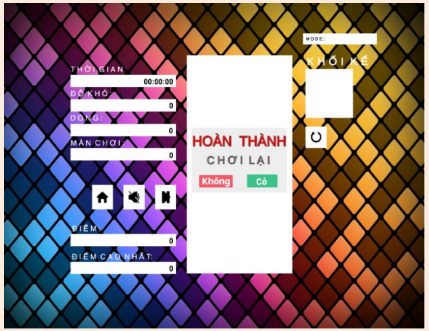


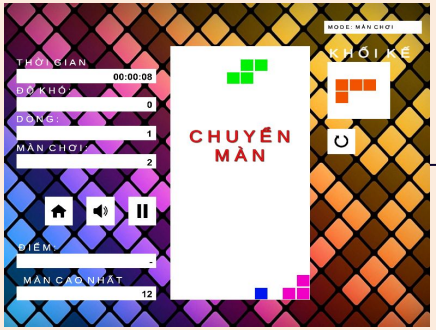
Giao diện Play Game

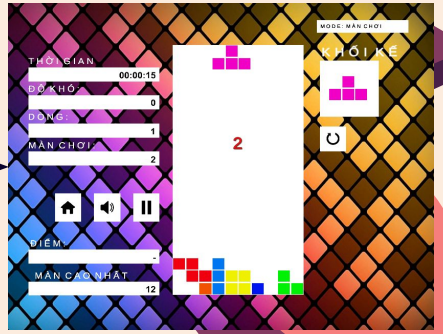






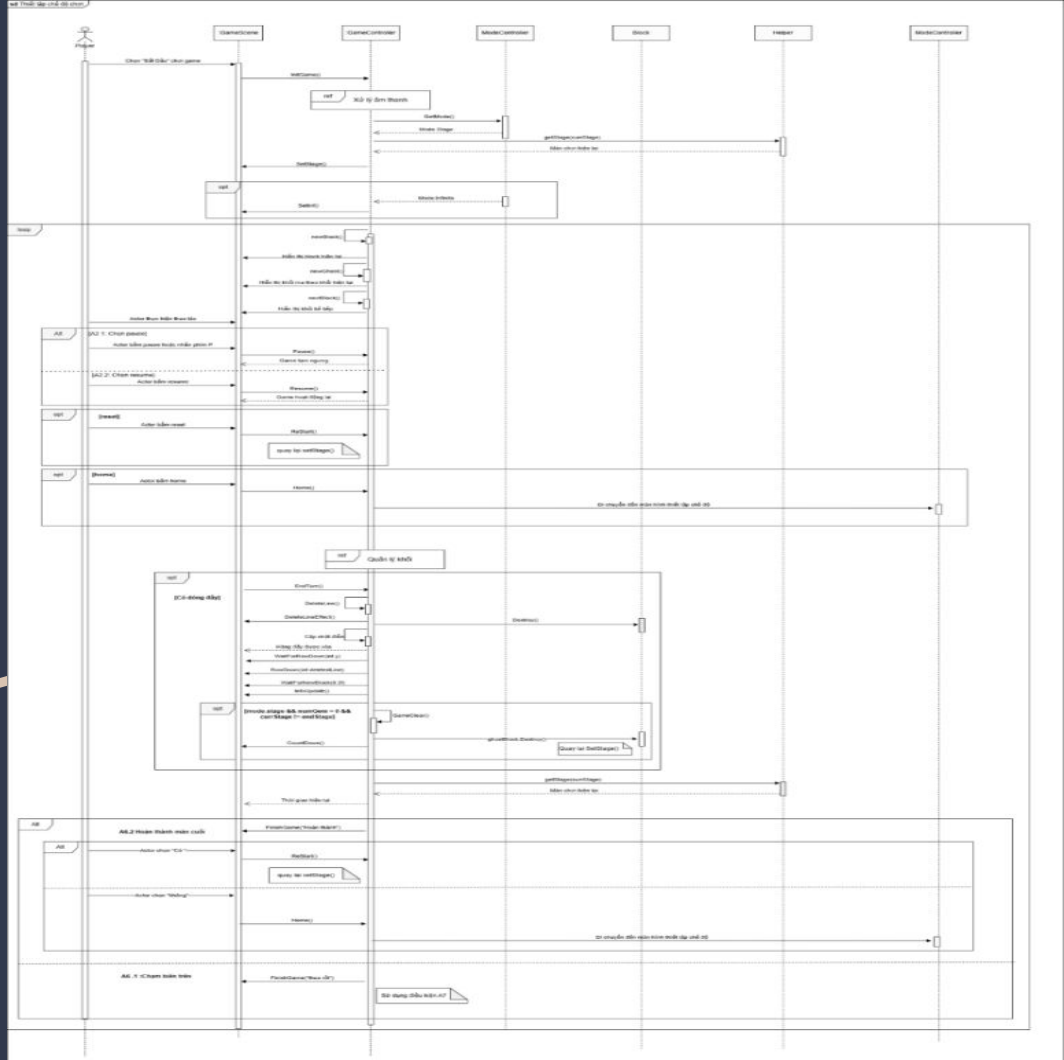






Phân tích yêu cầu

Sequence diagram
UC04



Thiết kế lớp và biểu đồ lớp

7 lớp chính

Sound.cs: Quản lý thông tin và phát âm thanh (tên, âm lượng, cao độ, chế độ lặp).

AudioManager.cs: Điều khiển toàn bộ hệ thống âm thanh (khởi tạo, phát và dừng âm thanh).

Block.cs: Class cơ sở cho các khối, quản lý điểm xoay, hiển thị và hủy khối.

GameController.cs: Quản lý trạng thái trò chơi, điều khiển khối, xử lý va chạm và UI.

GuideController.cs: Hướng dẫn người chơi mới, cung cấp thao tác quay lại màn hình cài đặt.

ModeController.cs: Quản lý và chuyển đổi chế độ chơi (vô cực, màn chơi).

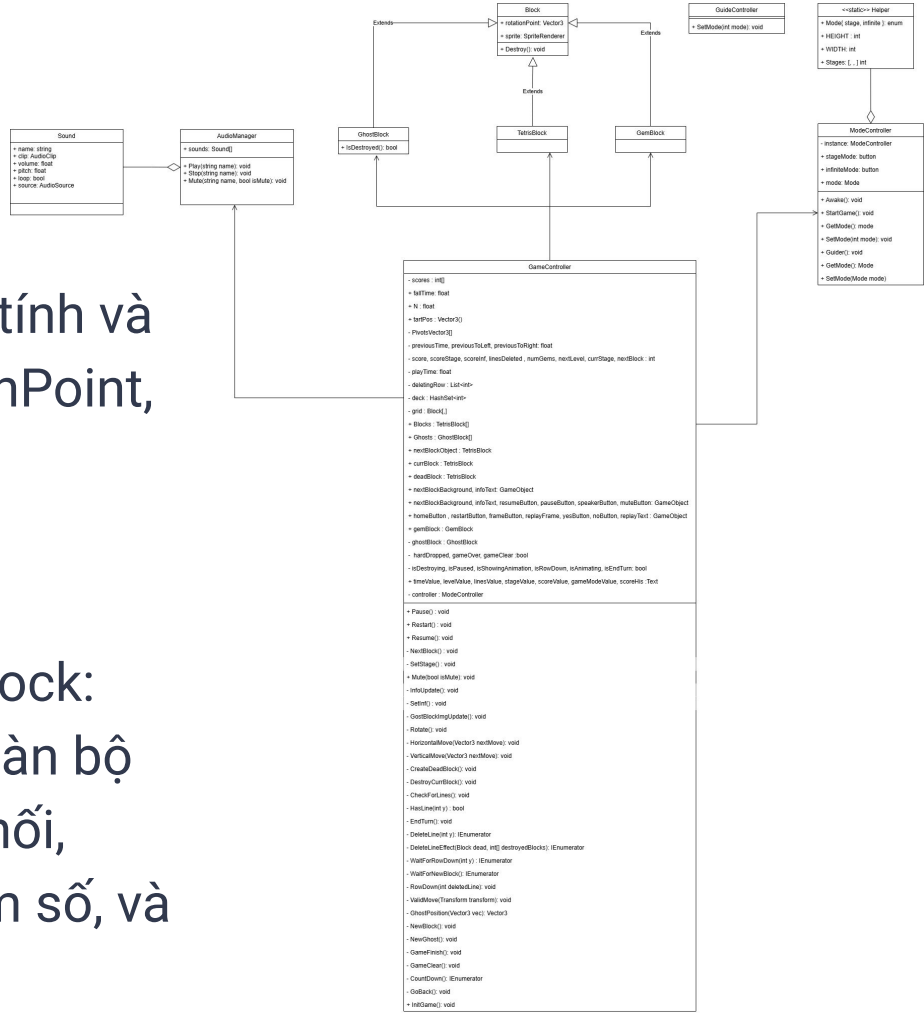
Helper.cs: Class tiện ích chứa hằng số và cấu hình trò chơi (kích thước lưới, chế độ).

Thiết kế lớp và biểu đồ lớp

- Do trong game sẽ có các đối tượng là gemBlock, Tetriminos, GhostBlock nên để dễ dàng quản lý và phát triển về sau, ta sẽ tạo ra các lớp con kết thừa từ Block :
  - GemBlock
  - GhostBlock
  - TetrisBlock
- Về bản chất, deadBlock không cần lớp riêng vì không có khác biệt đáng kể về thuộc tính hoặc hành vi so với TetrisBlock. Việc chuyển đổi trạng thái "đóng băng" có thể được xử lý bằng việc đánh dấu lại vị trí đó trên lưới game.

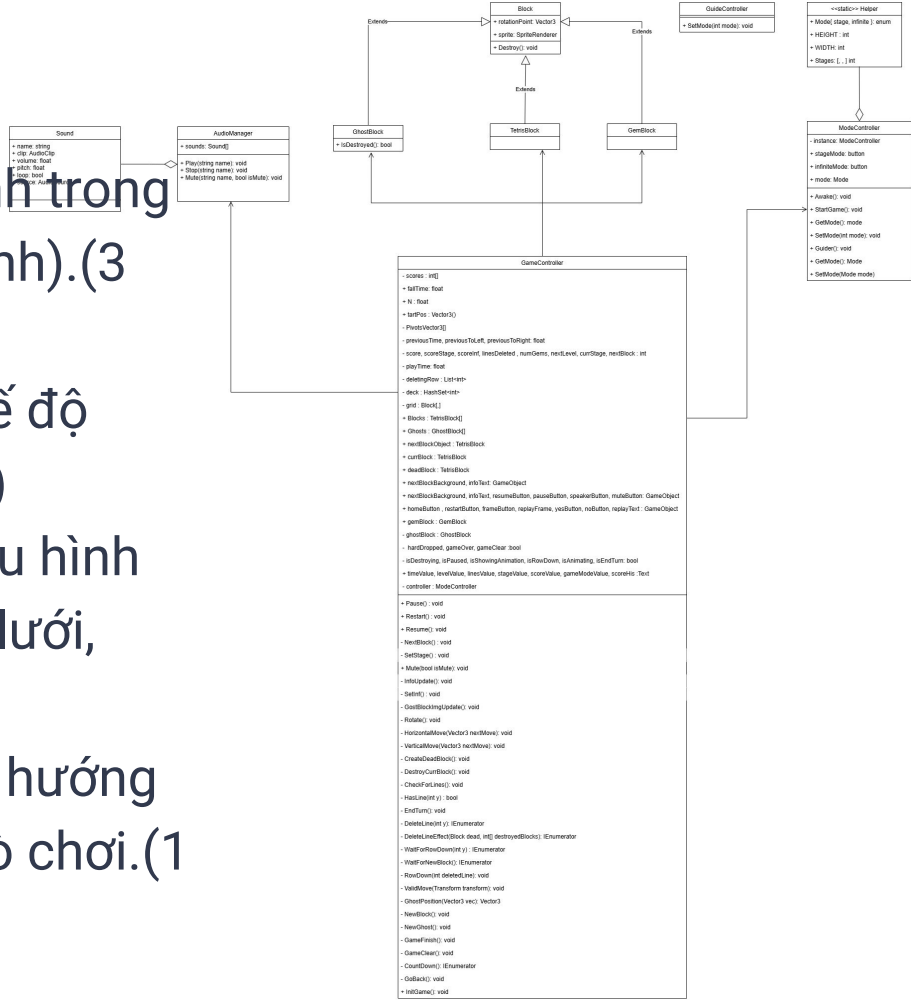
Thiết kế lớp và biểu đồ lớp

- **Lớp Block (Cơ sở):** Chứa thuộc tính và phương thức chung như rotationPoint, SpriteRenderer, và phương thức Destroy().(1 method)
- **Các lớp con kế thừa từ Block**: TetrisBlock, GemBlock, GhostBlock:
- **Lớp GameController:** Quản lý toàn bộ quá trình chơi, điều khiển các khối, kiểm tra va chạm, cập nhật điểm số, và kết thúc game. (31 methods)



Thiết kế lớp và biểu đồ lớp

- *Lớp AudioManager:* Quản lý âm thanh trong trò chơi (nhạc nền, hiệu ứng âm thanh).(3 methods)
- *Lớp ModeController:* Quản lý các chế độ chơi (vô hạn, giai đoạn).(7 methods)
- *Lớp Helper:* Chứa các hằng số và cấu hình quan trọng của trò chơi (kích thước lưới, thông tin màn chơi).
- *Lớp GuideController:* Quản lý chế độ hướng dẫn giúp người chơi làm quen với trò chơi.(1 method)



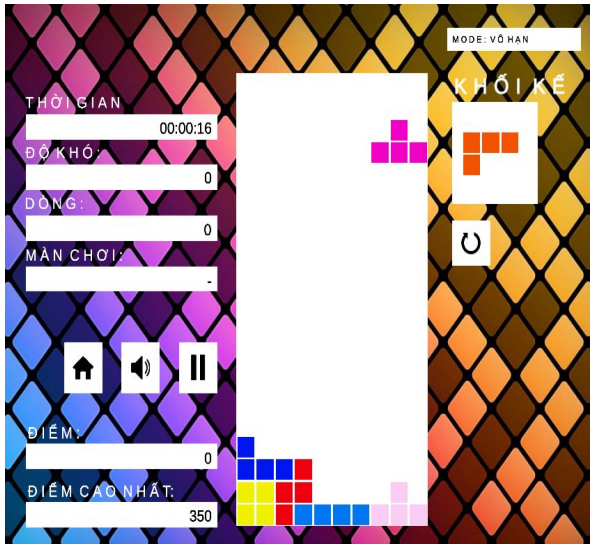
~Cách vận hành~

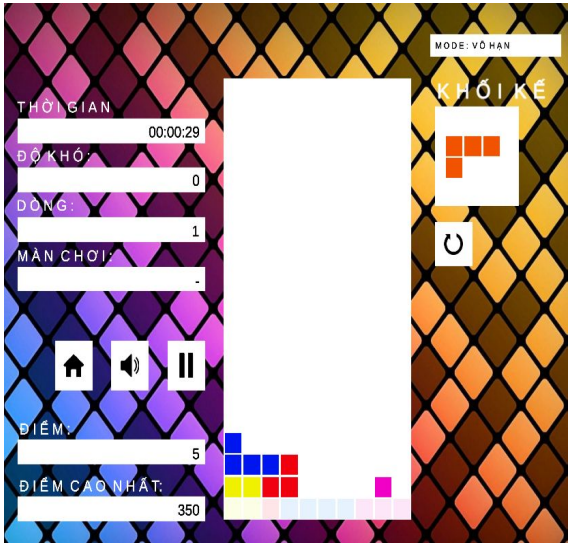
Khởi tạo trò chơi: Sử dụng GameController để khởi tạo lưới, khối và trạng thái ban đầu.
Điều khiển:

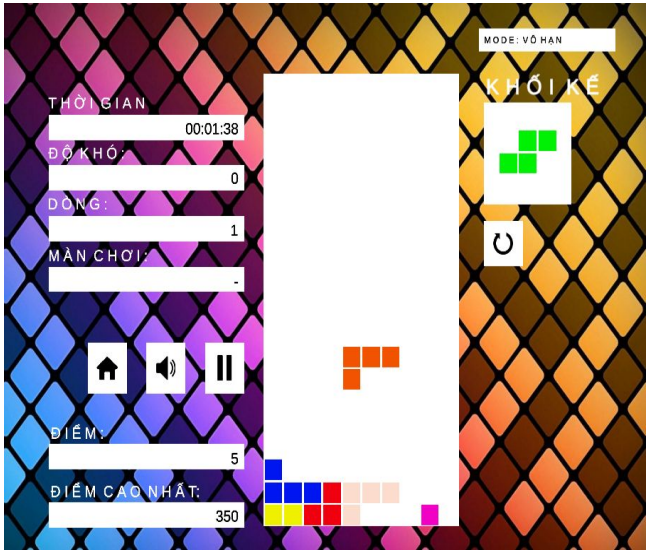
- Xoay khối (phím **Up**).
- Di chuyển trái/phải (phím **Left/Right**).
- Hạ nhanh khối (**Hard Drop**, phím **Space**).

Kiểm tra va chạm: Đảm bảo khối không vượt lưới hoặc chồng lên khối khác.
Xóa hàng: Hàng lấp đầy được xóa và các khối trên rơi xuống.
Chuyển chế độ: Hỗ trợ chơi theo màn và chế độ vô hạn.

Quy trình xử lý xóa hàng - cập nhật điểm







Các thư viện và công nghệ sử dụng

```
while (_progress > 0.0f)
{
    dead.sprite.GetComponent<SpriteRenderer>().color
        = new Color(tmp.r, tmp.g, tmp.b, tmp.a * _progress);
    _progress -= 0.1f;
    yield return new WaitForSeconds(0.03f);
}
```

- **UnityEngine:** Quản lý các đối tượng trong trò chơi (GameObject, Transform).
- **Unity UI**: Xây dựng giao diện HUD (TextMeshPro, Buttons).
- **System.Collections và System:** Xử lý vòng lặp trò chơi, sự kiện, và dữ liệu.
- **PlayerPrefs:** Lưu trữ điểm số và tiến trình.
- **Random:** Sinh khối Tetrimino ngẫu nhiên.
- **SpriteRenderer** (*component trong Unity*) : chỉnh sửa màu sắc, độ mờ của đối tượng
- **Coroutine** trong Unity để trì hoãn thời gian giúp thực hiện hiệu ứng thay đổi chậm hơn

```
if ((currStage + 1) >= scoreStage)
{
    scoreHis.text = (currStage + 1).ToString();

    // Lưu điểm số
    PlayerPrefs.SetInt("ScoreStage", (currStage + 1));
    PlayerPrefs.Save();
}
else scoreHis.text = scoreStage.ToString();
```

```
3 references
void NextBlock()
{
    print("nextblock start");
    if (deck.Count == Blocks.Length) deck.Clear();
    do nextBlock = Random.Range(0, Blocks.Length);
    while (deck.Contains(nextBlock));
    deck.Add(nextBlock);
```

V. Đánh giá và hướng phát triển

~Đánh giá~

- Lối chơi đơn giản nhưng đầy thách thức
- Rèn luyện khả năng tư duy chiến lược và phản xạ nhanh
- Tính dễ tiếp cận và phù hợp với tất cả mọi lứa tuổi

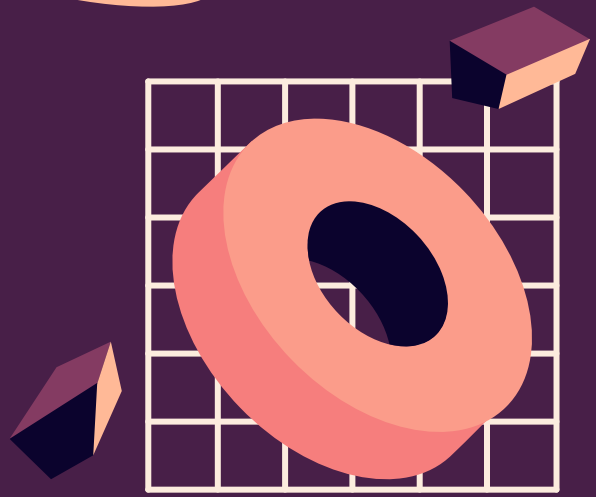
V. Đánh giá và hướng phát triển

~Hướng phát triển~

- **Mở rộng chế độ chơi.**
  + **Chạy đua**

  Mục tiêu: Xóa một số lượng dòng nhất định trong thời gian ngắn nhất
  + **Siêu tốc**

  Mục tiêu: Hoàn thành một số lượng dòng trong thời gian ngắn nhất.
- **Thêm tính năng Multiplayer**



Thank you!

Do you have any questions?