KHOA CÔNG NGHỆ THÔNG TIN

BỘ MÔN CÔNG NGHỆ PHẦN MỀM

Icon

Description automatically generated

ĐỒ ÁN MÔN HỌC

**Thiết kế kiến trúc phần mềm**

**Đề tài: Game nối điểm**

Nhóm sinh viên thực hiện: Nhóm 16

Danh sách sinh viên: **Dương Văn Nghĩa <0198466>**

**Nguyễn Đức Tâm <0203766>**

Lớp học phần: **66PM1**

Giảng viên hướng dẫn:  **Phạm Hữu Tùng**

Hà Nội 5/2025

Mục lục

**Chương 1. Giới thiệu đề tài**

1.1. Lý do chọn đề tài  
1.2. Mục tiêu của đồ án  
1.3. Phạm vi và giới hạn  
1.4. Phương pháp thực hiện

**Chương 2. Phân tích hệ thống**

2.1. Mô tả bài toán  
2.2. Yêu cầu chức năng  
2.3. Yêu cầu phi chức năng  
2.4. Use Case Diagram tổng quát  
2.5. Mô tả các Use Case chính  
2.6. Biểu đồ hoạt động (*Các chức năng chính*)

**Chương 3. Thiết kế hệ thống**

3.1. Lựa chọn mô hình kiến trúc

3.2. Sơ đồ kiến trúc tổng thể  
3.3. Biểu đồ tuần tự

3.4. Thiết kế lớp (Class Diagram)  
3.5. Thiết kế cơ sở dữ liệu

3.6. Thiết kế giao diện người dùng   
**Chương 4. Cài đặt và triển khai hệ thống**

4.1. Môi trường triển khai  
4.2. Cài đặt hệ thống

4.2.1 Yêu cầu hệ thống

4.2.2 Các bước cài đặt chương trình  
**Chương 5. Kết quả và đánh giá**

5.1. Kết quả thử nghiệm hệ thống  
5.2. Đánh giá hiệu quả hệ thống  
**Tài liệu tham khảo**

**Phụ lục (nếu có)**

**LỜI MỞ ĐẦU**

Đầu tiên, Nhóm 16 xin cảm ơn thầy Phạm Hữu Tùng đã góp ý, giúp đỡ và chỉ bảo trong suốt quá trình thực hiện đồ án môn học “Thiết kế kiến trúc phần mềm” với đề tài “**Đồ án Game: Connect Puzzle**”. Trong quá trình thực hiện đề tài cũng như phần bảo vệ, Nhóm 16 đã cố gắng trong khả năng, tuy nhiên sẽ vẫn còn những mặt thiếu sót, rất mong nhận được sự góp ý của thầy. Chân thành cảm ơn

**Nhóm 16 -  66PM1**

# **Chương I: Giới thiệu đề tài**

## 1.1 lý do chọn đề tài:

Việc lựa chọn phát triển một game giải đố ("Connect Puzzle") xuất phát từ niềm yêu thích đối với thể loại game đòi hỏi tư duy logic và mang tính thử thách. Đồng thời, dự án nhằm mục đích tạo ra một sản phẩm giải trí nhẹ nhàng, giúp người chơi thư giãn và rèn luyện trí não thông qua việc giải các câu đố nối điểm. Game giải đố nối điểm là một ý tưởng quen thuộc nhưng vẫn có nhiều không gian để sáng tạo và mang lại trải nghiệm mới mẻ cho người chơi.

## 1.2 Mục tiêu của đồ án game :

\* Tạo ra một game giải đố một người chơi offline với mục đích chính là giải trí.

\* Mang lại những giây phút thư giãn và thử thách trí tuệ cho người chơi thông qua việc giải các màn chơi nối điểm có độ khó tăng dần.

\* Hoàn thiện một sản phẩm game có thể chơi được với đầy đủ các tính năng cơ bản đã đặt ra.

## 1.3 Phạm vi và giới hạn :

Phạm vi:

\* Lối chơi: Game giải đố một người chơi, offline.

\* Cấu trúc game: Bao gồm 7 stage, mỗi stage có 50 level.

\* Kích thước bảng chơi: Thay đổi theo stage, cụ thể là `(số thứ tự stage + 4) x (số thứ tự stage + 4)`.

\* Số cặp điểm: Số cặp điểm cần nối trong mỗi stage là `(số thứ tự stage + 4)`.

\* Cơ chế chiến thắng: Nối hết tất cả các cặp điểm cùng màu với nhau sao cho các đường nối không cắt chéo nhau, chỉ đi ngang hoặc dọc, và lấp đầy toàn bộ các ô trên bảng chơi.

\* Tính năng sau khi thắng: Cho phép người chơi chọn chơi lại level hiện tại, chuyển sang màn chơi tiếp theo, hoặc trở về menu chính.

\* Lưu trữ tiến độ: Hệ thống lưu lại các level đã được người chơi hoàn thành.

\* Mở khóa màn chơi: Người chơi có thể chọn bất kỳ stage nào để bắt đầu. Level 1 của mỗi stage luôn được mở. Các level tiếp theo trong cùng một stage (từ 2 đến 50) sẽ được mở khóa tuần tự sau khi người chơi hoàn thành level liền trước.

\* Giao diện màn chơi: Hiển thị thông tin tên stage, số level hiện tại. Cung cấp các nút chức năng: "Chơi lại" (Restart), "Về Menu" (Back to Menu), và "Màn tiếp theo" (Next Level).

Giới hạn:

\*   Do hạn chế về thời gian phát triển, các tính năng nâng cao như cốt truyện phức tạp, đa dạng chế độ chơi, bảng xếp hạng trực tuyến, gợi ý (hint) chưa được triển khai trong phiên bản này.

\*   Game được phát triển dưới dạng offline, không có các tính năng tương tác mạng.

\*   Đồ họa và âm thanh được thiết kế ở mức độ đơn giản, tập trung vào trải nghiệm giải đố cốt lõi.

## 1.4 Phương pháp thực hiện :

\* Công cụ phát triển: Game được phát triển bằng Unity Engine.

\* Ngôn ngữ lập trình: C#.

\* Quy trình phát triển: (Có thể bổ sung nếu bạn theo một quy trình cụ thể như Agile, Waterfall, hoặc phát triển theo các cột mốc cá nhân).

# **Chương II: Phân tích hệ thống**

## 2.1 Mô tả bài toán :

Người chơi cần giải quyết bài toán tìm các đường nối để kết nối tất cả các cặp điểm (node) cùng màu cho trước trên một bảng chơi (grid) có kích thước giới hạn. Mục tiêu là nối thành công tất cả các cặp điểm sao cho các đường nối không cắt chéo hoặc đè lên nhau, chỉ được vẽ theo phương ngang hoặc phương dọc, và đồng thời tất cả các ô trên bảng chơi phải được lấp đầy bởi các đường nối này.

## 2.2 Yêu cầu chức năng :

\*   1: Quản lý Stage và Level

    \*   1.1: Hệ thống cho phép người chơi chọn bất kỳ stage nào trong tổng số 7 stage để bắt đầu chơi.

    \*   1.2: Khi một stage được chọn, level 1 của stage đó mặc định được mở và có thể chơi.

    \*   1.3: Các level tiếp theo (từ 2 đến 50) trong cùng một stage sẽ được mở khóa tuần tự khi người chơi hoàn thành level liền trước.

    \*   1.4: Hệ thống lưu trữ trạng thái hoàn thành của từng level.

\*   2: Hiển thị giao diện màn chơi (Gameplay UI)

    \*   2.1: Hiển thị bảng chơi (grid) với kích thước `(số thứ tự stage + 4) x (số thứ tự stage + 4)`.

    \*   2.2: Hiển thị các cặp điểm màu (nodes) trên bảng chơi, số lượng `(số thứ tự stage + 4)` cặp.

    \*   2.3: Hiển thị thông tin văn bản: "Tên Stage - Số Level" (ví dụ: "Stage 1 - Level 5").

    \*   2.4: Luôn hiển thị 3 nút điều khiển: "Chơi lại (Restart)", "Quay lại Menu (Back to Menu)", "Màn tiếp theo (Next Level)".

\*   FR3: Tương tác nối điểm

    \*   3.1: Khi người chơi chạm vào một điểm node ban đầu hoặc một ô đã được nối (thuộc một đường nối đang hình thành): Hệ thống hiển thị một "Highlight" hình tròn, có màu tương ứng với màu của node đó, kích thước highlight thay đổi theo stage.

    \*   3.2: Nếu người chơi chạm vào một ô trống không phải là điểm node hoặc không kề cạnh một đường nối đang vẽ, không có phản hồi nào.

    \*   3.3: Khi người chơi kéo để vẽ đường nối: Hệ thống hiển thị các "edge" (cạnh nối giữa các ô) theo màu của node ban đầu.

    \*   3.4: Khi người chơi hoàn thành việc nối một cặp điểm (điểm đầu và điểm cuối cùng màu gặp nhau): Hệ thống làm nổi bật các node của cặp đó.

\*   4: Kiểm tra và xử lý đường nối hợp lệ/không hợp lệ

    \*   4.1: Đường nối chỉ được vẽ theo chiều ngang hoặc dọc giữa các ô liền kề.

    \*   4.2: Đường nối không được tự cắt chính nó trong quá trình hình thành một đường dẫn đơn lẻ.

    \*   4.3: Đường nối không được cắt hoặc đè lên đường nối của một màu khác.

    \*   4.4: Xử lý tự động khi người chơi thực hiện thao tác vẽ không hợp lệ: Hệ thống sẽ tự động điều chỉnh hoặc xóa các đường nối hiện có để đảm bảo tính toàn vẹn của luật chơi, cụ thể:

        \*   Khi vẽ vào ô không liền kề: Thao tác không được thực hiện, đường nối không được hình thành tới ô đó.

        \*   Khi rút lại đường đang vẽ (vẽ đè lên đoạn vừa vẽ): Phần đường bị vẽ đè lên sẽ được xóa, cho phép người chơi "tẩy" đường đi.

        \*   Khi cố gắng tạo nhánh từ một ô trung gian đã có 2 kết nối (thuộc cùng một đường dẫn): Hệ thống sẽ tự động phá vỡ một trong các kết nối cũ của ô đó để tạo đường mới, ưu tiên giữ lại đường dẫn đang kết nối tới một điểm cuối (nếu có).

        \*   Khi vẽ vào một ô trung gian đã có 2 kết nối (thuộc cùng một đường dẫn): Các đường nối cũ của ô đó sẽ bị xóa để tạo đường mới, đảm bảo ô trung gian không có quá 2 kết nối cùng màu.

        \*   Khi vẽ vào một ô đã thuộc đường đi của màu khác: Đường đi của màu khác tại điểm đó sẽ bị xóa để giải phóng ô cho đường đi mới.

        \*   Khi vẽ từ/tới một điểm cuối đã có kết nối: Đường nối cũ của điểm cuối đó sẽ bị xóa, đảm bảo mỗi điểm cuối chỉ có một đường nối duy nhất.

        \*   Khi tạo vòng lặp kín cùng màu không hợp lệ (không nối tới điểm cuối hoặc tạo nhánh không đúng luật): Hệ thống sẽ tự động phá vỡ vòng lặp bằng cách xóa một số kết nối để duy trì tính hợp lệ của đường đi.

\*   5: Kiểm tra điều kiện thắng level

    \*   FR5.1: Hệ thống xác định người chơi thắng khi tất cả các cặp điểm đã được nối thành công VÀ tất cả các ô trên bảng chơi đã được lấp đầy bởi các đường nối.

\*   6: Chức năng các nút điều khiển trong màn chơi\*\*

    \*   6.1: Nút "Chơi lại (Restart)": Có thể nhấn bất kỳ lúc nào. Khi nhấn, xóa tất cả các đường nối đã vẽ, đặt lại bảng chơi về trạng thái ban đầu của level.

    \*   6.2: Nút "Quay lại Menu (Back to Menu)": Có thể nhấn bất kỳ lúc nào. Khi nhấn, đưa người chơi trở về màn hình chọn Stage/Menu chính.

    \*   6.3: Nút "Màn tiếp theo (Next Level)": Chỉ có thể nhấn được sau khi người chơi đã hoàn thành level hiện tại. Khi nhấn, tải và bắt đầu level tiếp theo.

## 2.3 Yêu cầu phi chức năng :

\*   Tính dễ sử dụng (Usability): Giao diện người dùng (UI) trực quan, dễ hiểu. Thao tác nối điểm đơn giản, mượt mà, có phản hồi rõ ràng.

\*   Hiệu năng (Performance): Game chạy ổn định, không giật lag, đặc biệt với các bảng chơi lớn. Thời gian tải level nhanh chóng.

\*   Tính ổn định (Stability): Game hoạt động tin cậy, không xảy ra lỗi nghiêm trọng (crash) làm gián đoạn trải nghiệm người dùng.

\*   Đồ họa (Graphics): Đồ họa 2D đơn giản, rõ ràng. Màu sắc được sử dụng để phân biệt tốt các cặp điểm và đường nối, dễ nhìn.

\*   Âm thanh (Sound): Có các hiệu ứng âm thanh cơ bản phản hồi các thao tác của người chơi (ví dụ: khi nối điểm, nối sai, hoàn thành level).

## 2.4 Use case Diagram tổng quát :

A diagram of a person with text

AI-generated content may be incorrect.

## 2.5 Mô tả các use case chính

**UC01: Start**

| **ID** | **UC01** |
| --- | --- |
| **Name** | **Start** |
| **Brief description** | **Người chơi bắt đầu truy cập vào game nối điểm.** |
| **Actors** | **Người chơi** |
| **Pre-conditions** | **Ứng dụng/game đã cài đặt thành công.- Người chơi đã mở ứng dụng/game.** |
| **Basic flow** | **1. Người chơi mở game. 2. Hệ thống hiển thị màn hình chào mừng và các lựa chọn tiếp theo (chọn stage, chọn level, v.v.).** |
| **Alternative flow** |  |
| **Result** | **Người chơi truy cập vào giao diện chính của game.** |
| **Exceptions** |  |

**UC02: Chọn Stage**

| **ID** | **UC02** |
| --- | --- |
| **Name** | **Chọn Stage** |
| **Brief description** | **Cho phép người chơi chọn stage muốn chơi.** |
| **Actors** | **Người chơi** |
| **Pre-conditions** | **- Đã truy cập vào giao diện chính của game. - Danh sách stage đã được tải.** |
| **Basic flow** | **1. Hệ thống hiển thị danh sách các stage. 2. Người chơi chọn stage mong muốn. 3. Hệ thống chuyển sang giao diện chọn level của stage đó.** |
| **Alternative flow** |  |
| **Result** | **Người chơi chọn được stage để tiếp tục chọn level.** |
| **Exceptions** |  |

**UC03: Chọn Level**

| **ID** | **UC03** |
| --- | --- |
| **Name** | **Chọn Level** |
| **Brief description** | **Cho phép người chơi chọn level muốn chơi trong một stage.** |
| **Actors** | **Người chơi** |
| **Pre-conditions** | **- Đã chọn stage. - Danh sách level của stage đã được tải.** |
| **Basic flow** | **1. Hệ thống hiển thị danh sách các level trong stage. 2. Người chơi chọn level mong muốn. 3. Hệ thống chuyển sang giao diện chơi màn chơi đó.** |
| **Alternative flow** |  |
| **Result** | **Người chơi chọn được level để bắt đầu chơi.** |
| **Exceptions** |  |

**UC04: Chơi màn chơi**

| **ID** | **UC03** |
| --- | --- |
| **Name** | **Chơi màn chơi** |
| **Brief description** | **Người chơi thực hiện các thao tác nối điểm, hoàn thành thử thách của level.** |
| **Actors** | **Người chơi** |
| **Pre-conditions** | **- Đã chọn level thành công. - Dữ liệu màn chơi đã được tải.** |
| **Basic flow** | **1. Hệ thống hiển thị bảng chơi. 2. Người chơi thao tác nối các điểm cùng màu. 3. Hệ thống kiểm tra điều kiện thắng. 4. Nếu thắng, hiển thị thông báo và các lựa chọn tiếp theo.** |
| **Alternative flow** | **- Nếu thao tác không hợp lệ (nối sai quy tắc), hệ thống không thực hiện. - Nếu người chơi muốn dừng, có thể chọn về menu chính.** |
| **Result** | **Người chơi hoàn thành màn chơi hoặc quay lại menu.** |
| **Exceptions** |  |

## 2.6 Biểu đồ hoạt động:

**a. Biểu đồ hoạt động: Chơi game**

**Ảnh có chứa văn bản, ảnh chụp màn hình, biểu đồ, Song song

Nội dung do AI tạo ra có thể không chính xác.**

**b. Biểu đồ hoạt động: Chọn màn chơi**

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, số

Nội dung do AI tạo ra có thể không chính xác.

**c. Biểu đồ hoạt động: Âm thanh/nhạc nền**

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, hàng

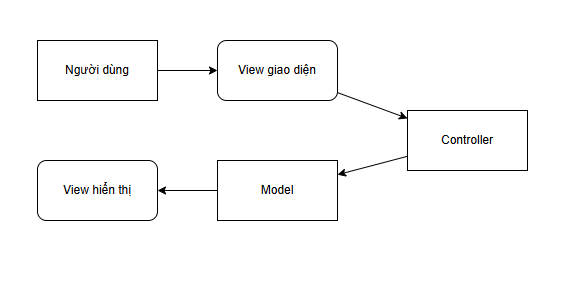
Nội dung do AI tạo ra có thể không chính xác.

# **Chương III: Thiết kế hệ thống**

## 3.1 Lựa chọn mô hình kiến trúc :

Dự án sử dụng mô hình Kiến trúc dựa trên Thành phần (Component-Based Architecture), một mô hình phổ biến và hiệu quả trong phát triển game với Unity Engine. Trong mô hình này, các đối tượng trong game (GameObjects) được cấu thành từ nhiều thành phần (Components) độc lập, mỗi thành phần đảm nhiệm một chức năng hoặc hành vi cụ thể. Các script C# kế thừa từ `MonoBehaviour` chính là các component này. Unity quản lý vòng đời và sự tương tác của chúng.

## 3.2 Sơ đồ kiến trúc tổng thể :



## 3.3 Biểu đồ tuần tự :

1. Bắt đầu trò chơi

A screenshot of a computer

AI-generated content may be incorrect.

2. Chọn Stage

A screenshot of a computer

AI-generated content may be incorrect.

3. Chọn level

A screenshot of a computer

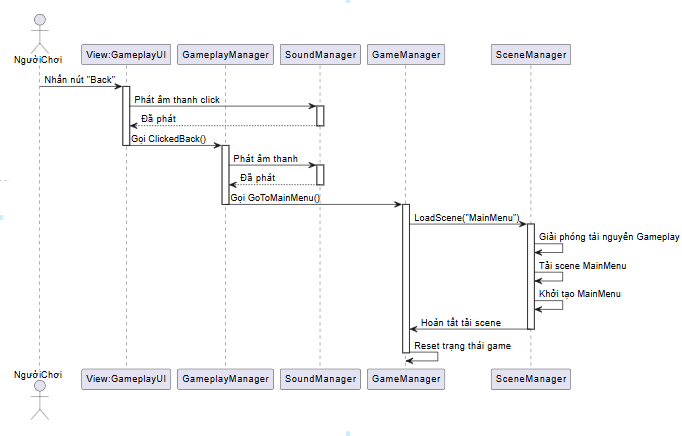
AI-generated content may be incorrect.

4. Nối điểm

A computer screen shot of a diagram

AI-generated content may be incorrect.

5. Nhấn nút Back



6. Nhấn nút Restart

A diagram of a game

AI-generated content may be incorrect.

7. Nhấn nút NextLevel

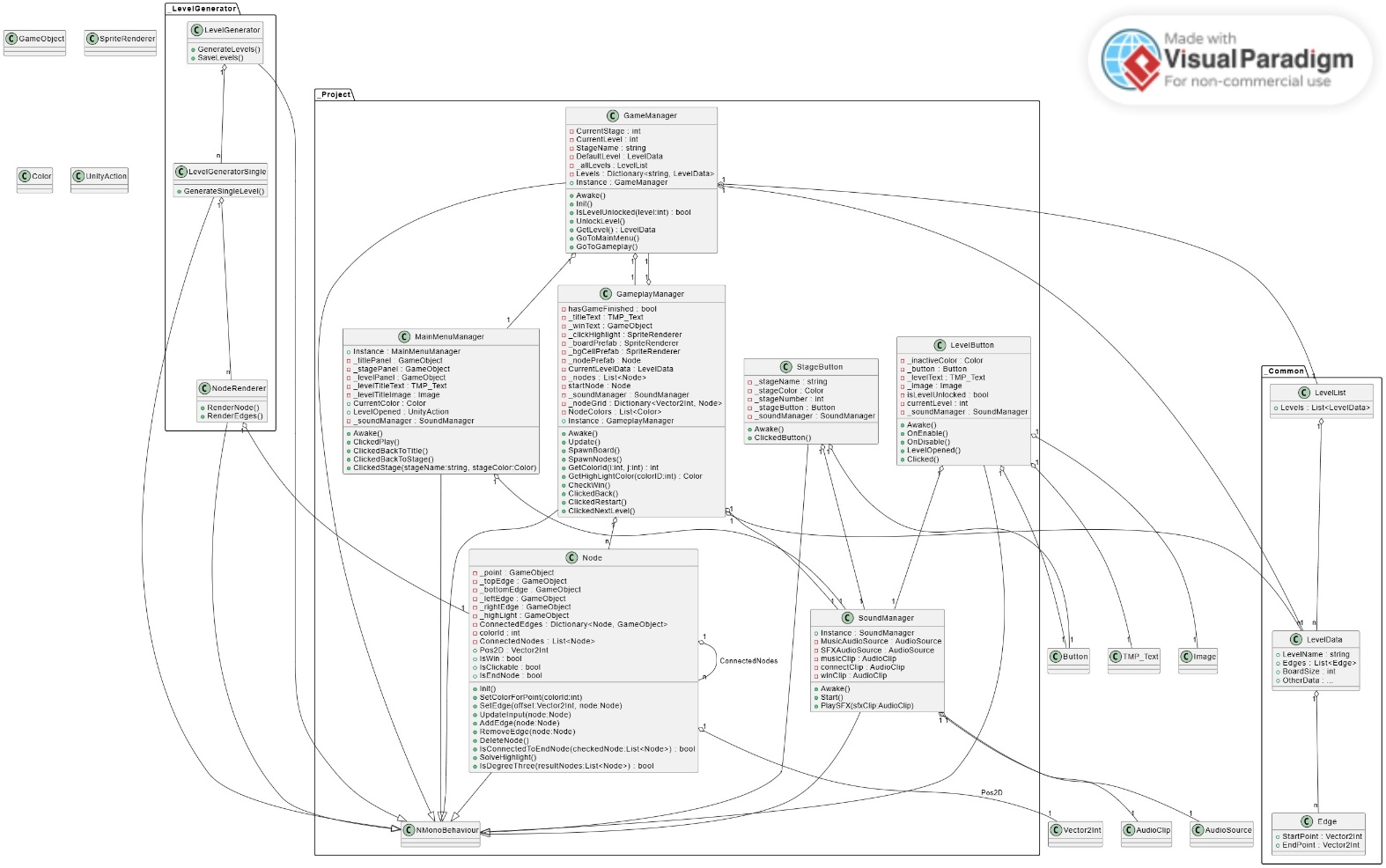
A diagram of a project

AI-generated content may be incorrect.

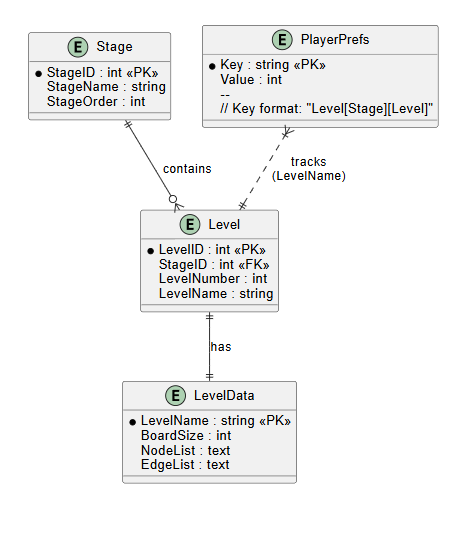
7. Khi hoàn thành levelA screenshot of a computer program

AI-generated content may be incorrect.

## 3.4 Thiết kế lớp (Class Diagram) :

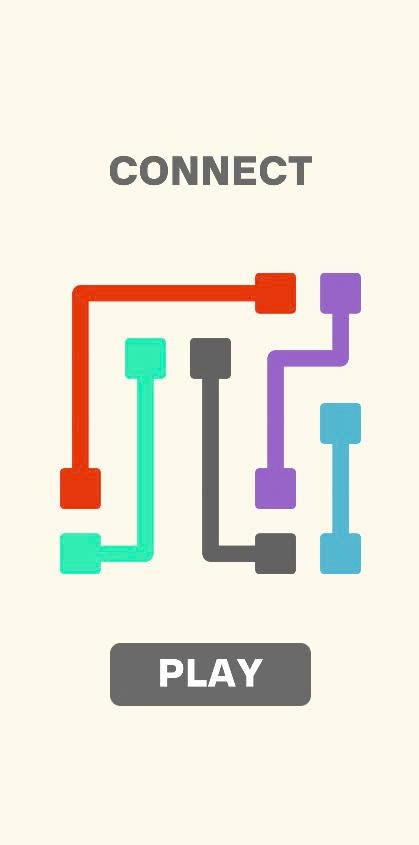


## 3.5 Thiết kế cơ sở dữ liệu :



## 3.6 Thiết kế giao diện người dùng :

+Màn hình 1: Bắt đầu game



+Màn hình 2: Màn hình hiển thị Stage

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, Nhiều màu sắc

Nội dung do AI tạo ra có thể không chính xác.

+Màn hình 3 : Màn hình hiển thị các Level trong Stage

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, máy tính tay

Nội dung do AI tạo ra có thể không chính xác.

+Màn hình 4 : Màn hình hiển thị phần chơi Level

Ảnh có chứa hình vuông, Hình chữ nhật, ảnh chụp màn hình

Nội dung do AI tạo ra có thể không chính xác.

+Màn hình 5 : Màn hình khi hoàn thành phần chơi trong Level

Ảnh có chứa văn bản, ảnh chụp màn hình, Hình chữ nhật, Phông chữ

Nội dung do AI tạo ra có thể không chính xác.

# **Chương IV: Cài đặt và triển khai hệ thống**

## 4.1 Môi trường triển khai :

**\*   Engine: Unity 2022.3.46f1**

**\*   Ngôn ngữ lập trình: C# (phiên bản đi kèm với Unity)**

**\*  Nền tảng mục tiêu: Windows (PC) và Android.**

**\*   IDE (Môi trường phát triển tích hợp): Microsoft Visual Studio (khuyến nghị) hoặc Visual Studio Code (với các extension hỗ trợ Unity).**

## 4.2 Cài đặt hệ thống :

4.2.1 Yêu cầu hệ thống :

**Đối với nền tảng Windows (PC):**

**\*  Yêu cầu tối thiểu:**

**\*   Hệ điều hành: Windows 7 SP1+ (64-bit)**

**\*   CPU: Bộ xử lý hỗ trợ tập lệnh SSE2**

**\*   RAM: 4 GB**

**\*   Card đồ họa: Card đồ họa có khả năng DirectX 10.**

**\*   Dung lượng ổ cứng trống: Khoảng 500 MB (có thể thay đổi tùy theo số lượng assets cuối cùng).**

**\*   Yêu cầu đề nghị:**

**\*   Hệ điều hành: Windows 10/11 (64-bit)**

**\*   CPU: Intel Core i3 hoặc AMD Ryzen 3 trở lên.**

**\*   RAM: 8 GB**

**\*   Card đồ họa: Card đồ họa hỗ trợ DirectX 11 hoặc 12 với 1GB VRAM.**

**\*   Dung lượng ổ cứng trống: 1 GB.**

**Đối với nền tảng Android:**

**\*   Yêu cầu tối thiểu:**

**\*   Hệ điều hành: Android 5.1 (Lollipop) API Level 22 trở lên.**

**\*   CPU: ARMv7 với Neon Support hoặc ARM64.**

**\*   RAM: 2 GB.**

**\*   Dung lượng lưu trữ trống: Khoảng 200 MB.**

**\*   Yêu cầu đề nghị:**

**\*   Hệ điều hành: Android 8.0 (Oreo) API Level 26 trở lên.**

**\*   RAM: 3 GB trở lên.**

4.2.2 Các bước cài đặt chương trình/chi tiết triển khai logic :

**\*\*Cấu trúc thư mục chính trong `Assets/\_Project/`:\*\***

**\*   `\_Project/`**

**\*   `Prefabs/`: Chứa các Prefab của game như `NodePrefab`, các nút bấm UI, các phần tử giao diện khác.**

**\*   `Scenes/`: Chứa các file Scene của game, ví dụ: `MainMenuScene`, `StageSelectionScene`, `GameplayScene`.**

**\*   `Scripts/`: Chứa toàn bộ mã nguồn C# của dự án. Các script chính bao gồm:**

**\*   `GameManager.cs`: Quản lý trạng thái chung của game, lưu/tải dữ liệu, điều hướng scene.**

**\*   `MainMenuManager.cs`: Xử lý logic cho màn hình menu chính và chọn stage.**

**\*   `GameplayManager.cs`: Quản lý logic cốt lõi của màn chơi, tạo lưới, xử lý input, kiểm tra điều kiện thắng.**

**\*   `Node.cs`: Đại diện cho từng ô trên lưới, xử lý logic nối điểm và cập nhật hiển thị.**

**\*   `LevelButton.cs`: Script cho các nút chọn level.**

**\*   `StageButton.cs`: Script cho các nút chọn stage.**

**\*   `SoundManager.cs`: Quản lý việc phát âm thanh và nhạc nền.**

**\*   `Connect.Core.asmdef`: Assembly Definition File giúp tổ chức và tối ưu hóa thời gian biên dịch code.**

**\*   *(Các thư mục khác như `Art/`, `Audio/`, `Fonts/`, `ScriptableObjects/` có thể được thêm vào tùy theo nhu cầu phát triển để chứa hình ảnh, âm thanh, font chữ và các ScriptableObject định nghĩa dữ liệu level/stage nếu cần.)***

**Hướng dẫn Build Game:**

**1. Build cho Windows (PC):**

**1.  Mở dự án "Connect Puzzle" bằng Unity Hub với phiên bản Unity 2022.3.46f1.**

**2.  Trong Unity Editor, đi đến `File > Build Settings...`.**

**3.  Trong cửa sổ `Build Settings`:**

**\*   Platform: Chọn `Windows, Mac, Linux`.**

**\*   Target Platform: Chọn `Windows`.**

**\*   Architecture: Chọn `x86\_64`.**

**4.  Nhấn nút `Switch Platform` nếu nền tảng hiện tại không phải là Windows.**

**5.  Scenes In Build: Đảm bảo các Scene cần thiết (`MainMenuScene`, `GameplayScene`, v.v.) đã được thêm vào và sắp xếp đúng thứ tự (Scene đầu tiên sẽ là scene khởi động).**

**6.  (Tùy chọn) Nhấn `Player Settings...` để cấu hình tên công ty, tên sản phẩm, phiên bản, icon, độ phân giải mặc định, chế độ toàn màn hình, v.v.**

**7.  Nhấn nút `Build`.**

**8.  Chọn một thư mục để lưu trữ bản build và đặt tên cho file thực thi (e.g., `ConnectPuzzle.exe`).**

**9.  Chờ quá trình build hoàn tất.**

**\*\*2. Build cho Android:\*\***

**1.  Đảm bảo đã cài đặt Android Build Support trong Unity Hub cho phiên bản 2022.3.46f1 (bao gồm Android SDK & NDK Tools, OpenJDK).**

**2.  Trong Unity Editor, đi đến `File > Build Settings...`.**

**3.  Trong cửa sổ `Build Settings`:**

**\*   \*\*Platform:\*\* Chọn `Android`.**

**4.  Nhấn nút `Switch Platform`.**

**5.  \*\*Scenes In Build:\*\* Tương tự như build cho Windows.**

**6.  Nhấn `Player Settings...` và cấu hình các mục quan trọng cho Android:**

**\*   `Identification`:**

**\*   `Package Name`: Đặt tên định danh duy nhất (e.g., `com.yourcompanyname.connectpuzzle`).**

**\*   `Version` và `Bundle Version Code`.**

**\*   `Minimum API Level` và `Target API Level` (tham khảo yêu cầu hệ thống).**

**\*   `Resolution and Presentation`: Cấu hình hướng màn hình (e.g., `Portrait` hoặc `Landscape`).**

**\*   `Publishing Settings`: Cấu hình Keystore để ký ứng dụng (cần thiết để phát hành lên Google Play Store). Tạo mới hoặc sử dụng Keystore hiện có.**

**7.  Kết nối thiết bị Android với máy tính (đã bật USB Debugging) hoặc sử dụng Android Emulator.**

**8.  Nhấn `Build And Run` để build và tự động cài đặt lên thiết bị/emulator, hoặc nhấn `Build` để chỉ tạo file APK.**

**9.  Nếu chỉ `Build`, chọn thư mục lưu file APK (e.g., `ConnectPuzzle.apk`).**

# **Chương V: Kết quả và đánh giá**



## 5.1 Kết quả thử nghiệm hệ thống :

## 5.2 Đánh giá hiệu quả hệ thống :

**\*   Ưu điểm:**

**\*   Lối chơi cuốn hút: Game giải đố nối điểm là một thể loại quen thuộc, dễ hiểu, dễ tiếp cận nhưng vẫn mang tính thử thách cao, phù hợp với nhiều đối tượng người chơi.**

**\*   Cấu trúc nội dung tốt: Việc chia thành 7 stage, mỗi stage 50 level với độ khó tăng dần (kích thước bảng và số cặp điểm tăng) tạo ra lộ trình thử thách liên tục, giữ chân người chơi.**

**\*   Giao diện người dùng trực quan: Thiết kế UI (dựa trên mô tả Chương 3.6) hướng đến sự đơn giản, rõ ràng, giúp người chơi dễ dàng thao tác và tập trung vào việc giải đố.**

**\*   Cơ chế xử lý đường nối thông minh: Việc tự động xử lý các thao tác vẽ không hợp lệ (FR4.4) giúp giảm sự khó chịu cho người chơi và duy trì tính toàn vẹn của luật chơi.**

**\*   Hoạt động ổn định: Game được thiết kế để hoạt động ổn định trên các nền tảng mục tiêu (Windows, Android), đảm bảo trải nghiệm chơi game mượt mà.**

**\*   Lưu trữ tiến độ hiệu quả: Sử dụng `PlayerPrefs` là giải pháp đơn giản và phù hợp cho game offline một người chơi, giúp lưu lại tiến trình của người dùng một cách tin cậy.**

**\*   Nhược điểm:**

**\*   Đồ họa và Âm thanh: Hiện tại, đồ họa và âm thanh được mô tả ở mức đơn giản. Để tăng tính hấp dẫn, cần đầu tư thêm vào mặt hình ảnh (art style độc đáo, hiệu ứng đẹp mắt) và âm thanh (nhạc nền đa dạng, SFX sống động).**

**\*   Thiếu các tính năng hỗ trợ: Game có thể thiếu các tính năng thường thấy ở thể loại giải đố như gợi ý (hints), hoàn tác (undo), điều này có thể gây khó khăn cho người chơi ở những level phức tạp.**

**\*   Tính chơi lại (Replayability): Sau khi hoàn thành tất cả các level, động lực để người chơi quay lại có thể không cao nếu không có thêm các yếu tố mới (ví dụ: chế độ chơi tính thời gian, bảng xếp hạng, thử thách hàng ngày).**

**\*   Ít yếu tố bất ngờ/đột phá: Lối chơi nối điểm khá phổ biến, nếu không có thêm các cơ chế độc đáo hoặc biến thể mới lạ, game có thể khó nổi bật giữa các tựa game cùng thể loại.**

**\*   \*\*Hướng phát triển tương lai:\*\***

**\*   Nâng cấp Đồ họa và Âm thanh: Thiết kế lại giao diện với phong cách nghệ thuật đặc sắc hơn. Thêm các hiệu ứng hình ảnh khi nối điểm, hoàn thành level. Soạn nhạc nền và SFX chất lượng cao, phù hợp với không khí game.**

**\*   Bổ sung Tính năng Hỗ trợ:**

**\*   Thêm hệ thống gợi ý (ví dụ: hiển thị một phần đường nối đúng).**

**\*   Thêm chức năng hoàn tác (undo) một vài bước đi trước đó.**

**\*   Mở rộng Nội dung:**

**\*   Thiết kế thêm nhiều Stage và Level với độ khó và thử thách đa dạng hơn.**

**\*   Giới thiệu các cơ chế chơi mới: ô cản (blocker), ô một chiều, giới hạn số bước đi, nối nhiều hơn 2 điểm cùng màu.**

**\*   Tăng Tính Chơi Lại:**

**\*   Chế độ chơi tính thời gian (Time Attack).**

**\*   Hệ thống tính điểm dựa trên thời gian hoàn thành hoặc số bước ít nhất.**

**\*   Thử thách hàng ngày/hàng tuần với các level đặc biệt.**

**\*   Bảng xếp hạng (Leaderboards) nếu có kế hoạch phát triển tính năng online.**

**\*   Tùy chỉnh Giao diện: Cho phép người chơi thay đổi chủ đề (theme) của game (màu sắc, hình ảnh nền, kiểu đường nối).**

**\*   Đa ngôn ngữ: Hỗ trợ thêm nhiều ngôn ngữ khác nhau để tiếp cận thị trường rộng hơn.**

**\*   (Tùy chọn) Tích hợp Quảng cáo/IAP:\*\* Nếu có mục tiêu thương mại hóa, có thể xem xét tích hợp quảng cáo (ví dụ: quảng cáo để nhận gợi ý) hoặc mua vật phẩm trong game (IAP) một cách hợp lý để không ảnh hưởng tiêu cực đến trải nghiệm người chơi.**

# **Tài liệu tham khảo:**

# **Phụ lục (nếu có) :**