|  |
| --- |
| **­­­HỌC VIỆN HÀNG KHÔNG VIỆT NAM**  **KHOA CÔNG NGHỆ THÔNG TIN**      **BÁO CÁO**  **LẬP TRÌNH GAME**  **GAME PHIÊU LƯU**  **Giảng viên hướng dẫn: ThS. Huỳnh Thanh Sơn**  **Sinh viên/ Nhóm sinh viên thực hiện: Nhóm 04**  **Mã số sinh viên:**  **Lớp: 010100086501**  **TP.Hồ Chí Minh, tháng 07/2025** |
| **HỌC VIỆN HÀNG KHÔNG VIỆT NAM**  **KHOA CÔNG NGHỆ THÔNG TIN**      **BÁO CÁO**  **LẬP TRÌNH GAME**  **GAME PHIÊU LƯU**  **Giảng viên hướng dẫn: ThS. Huỳnh Thanh Sơn**  **Sinh viên/ Nhóm sinh viên thực hiện: Nhóm 04**  **Mã số sinh viên:**  **Lớp: 010100086501**  **Thành phố Hồ Chí Minh, tháng 07/2025** |

**Danh sách Nhóm:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Họ và tên** | **MSSV** | **Lớp** | **Ghi chú** |
| 1 | **Trần Phạm Minh Đức** | **2331540141** | **23ĐHTT03** | Nhóm Trưởng |
| 2 | **Lương Thế Phong** | **2331540161** | **23ĐHTT03** |  |
| 3 | **Đặng Hoàng Nguyên** | **2331540155** | **23ĐHTT03** |  |
| 4 | **Vũ Nguyên Long** | **2331540150** | **23ĐHTT03** |  |
| 5 | **Nguyễn Đức Việt Hùng** | **2331540130** | **23ĐHTT03** |  |

|  |  |
| --- | --- |
| **Cán bộ chấm thi 1**  *(ký và ghi rõ họ tên)* | **Cán bộ chấm thi 2**  *(ký và ghi rõ họ tên)* |
| **Cán bộ chấm thi phúc khảo 1**  *(ký và ghi rõ họ tên)* | **Cán bộ chấm thi phúc khảo 2**  *(ký và ghi rõ họ tên)* |

**DANH MỤC CÁC KÝ HIỆU, CHỮ VIẾT TẮT**

**(nếu có)**

|  |  |
| --- | --- |
| **Ký hiệu, chữ viết tắt** | **Chữ viết đầy đủ** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**DANH MỤC CÁC BẢNG, HÌNH VẼ, ĐỒ THỊ**

**(nếu có)**

**MỤC LỤC**

# *Trang*

# Danh mục các ký hiệu, chữ viết tắt

# Danh mục các bảng, hình vẽ, đồ thị

# **MỞ ĐẦU:**

## **CHƯƠNG 1: GIỚI THIỆU**

### **1.1. Lý do chọn đề tài**

### **1.2. Mục tiêu đề tài**

### **1.3 Phạm vi đề tài**

### **1.4 Đối tượng nghiên cứu**

### **1.5. Phương pháp nghiên cứu**

### **1.6. Bố cục đề tài**

## **CHƯƠNG 2: CƠ SỞ LÝ THUYẾT**

### **2.1. Unity Engine**

### **2.2. Ngôn ngữ lập trình C#**

### **2.3. Lưu lại bản Save sử dụng .JSON**

## **CHƯƠNG 3: PHÂN TÍCH VÀ XÂY DỰNG SẢN PHẨM GAME**

### **3.1. Tổng quan trò chơi**

### **3.1.2. Giới thiệu trò chơi**

### **3.1.3. Gameplay**

### **3.1.4. Cơ chế của trò chơi**

### **3.1.5. Điều khiển**

### **3.1.6. Cốt truyện**

### **3.1.7. Thế giới game**

### **3.1.8. Nhân vật và kẻ địch**

### **3.1.9. Màn chơi**

### **3.1.10. UI**

### **3.2. Phân tích hệ thống game**

### **3.2.1. Các tác nhân của hệ thống**

### **3.2.2. Chức năng hệ thống**

### **3.2.3. Biểu đồ use-case**

### **3.2.3. Biểu đồ lớp (Class Diagram)**

### **3.2.3. Biểu đồ hoạt động (Activity Diagram)**

### **3.2.4. Biểu đồ tuần tự (Sequence Diagram)**

### **3.2.5. Biểu đồ trạng thái (State Diagram)**

### **3.3. Xây dựng các phần thành của Game**

### **3.4. Xây dựng giao diện sản phẩm Game**

# **KẾT LUẬN**

# **DANH MỤC TÀI LIỆU THAM KHẢO**

# **PHỤ LỤC**

**MỞ ĐẦU**

Giới thiệu tổng quan về đề tài, viết ngắn ngọn, rõ nội dung.

**CHƯƠNG 1. GIỚI THIỆU**

**1.1. Lý do chọn đề tài**

Trong thời đại công nghệ 4.0, ngành công nghiệp trò chơi điện tử (game) đang phát triển mạnh mẽ và trở thành một trong những lĩnh vực giải trí có doanh thu cao nhất toàn cầu. Đặc biệt, với sự bùng nổ của điện thoại thông minh và nền tảng game đa nền, người dùng ngày càng có nhiều nhu cầu giải trí mọi lúc, mọi nơi.

Tại Việt Nam, số lượng người chơi game chiếm tỷ lệ cao trong dân số trẻ, cùng với xu hướng phát triển game nội địa ngày càng được quan tâm, khuyến khích. Tuy nhiên, việc tạo ra một trò chơi hấp dẫn và có khả năng mở rộng vẫn là một thách thức lớn đối với sinh viên công nghệ thông tin. Unity – một trong những nền tảng phát triển game phổ biến nhất hiện nay – cung cấp đầy đủ công cụ mạnh mẽ và miễn phí cho sinh viên, nhà phát triển độc lập để có thể xây dựng game 2D, 3D, đa nền tảng một cách dễ dàng.

Xuất phát từ thực tế đó, em chọn đề tài “Xây dựng trò chơi **LEVELING** sử dụng Unity” với mong muốn tiếp cận quy trình phát triển game từ lý thuyết đến thực hành, đồng thời áp dụng các kiến thức lập trình, xử lý đồ họa, vật lý, và tư duy logic đã học để xây dựng một ứng dụng giải trí trực quan, tương tác.

**1.2. Mục tiêu đề tài**

Đề tài hướng tới việc xây dựng một trò chơi [tên game] 2D/3D sử dụng nền tảng Unity, phục vụ nhu cầu giải trí cho người chơi, đặc biệt là đối tượng học sinh – sinh viên và người yêu thích thể loại game đơn giản, dễ tiếp cận.

Các mục tiêu cụ thể:

* **Về hệ thống ứng dụng**: Tạo ra một trò chơi có thể chạy được trên nền tảng Windows/Android, có giao diện thân thiện, điều khiển mượt mà, tích hợp các tính năng cơ bản như di chuyển nhân vật, tương tác vật thể, tính điểm, lưu tiến trình,...
* **Về công nghệ:** Nắm bắt và vận dụng các kiến thức về Unity, C#, Sprite, Animation, Prefab, Physics, UI System, cùng các kỹ thuật lưu trữ dữ liệu (PlayerPrefs/JSON).

**1.3. Phạm vi đề tài**

**Không gian**: Đề tài tập trung vào việc xây dựng game sử dụng công cụ Unity trong môi trường phát triển cục bộ (local) trên máy tính cá nhân.

**Thời gian**: Thực hiện trong khoảng thời gian một học kỳ.

**Nội dung và chức năng**: Trò chơi thuộc thể loại [Platformer / Arcade / RPG / Endless Runner...], ở mức độ cơ bản đến trung bình, có các tính năng như điều khiển nhân vật, tương tác với vật thể, xử lý va chạm, điểm số, giao diện người dùng (UI), lưu tiến trình. Không đi sâu vào AI, network multiplayer hoặc thị trường hóa.

**1.4 Đối tượng nghiên cứu**

Các đối tượng được nghiên cứu trong đề tài bao gồm:

* **Công cụ Unity**: giao diện, quy trình phát triển, hệ thống component.
* **Ngôn ngữ C# trong Unity**: dùng để lập trình hành vi cho các đối tượng trong game.
* **Các thành phần cơ bản của một game**: GameObject, Sprite, Rigidbody, Collider, UI, Prefab.

**1.5. Phương pháp nghiên cứu**

**Phương pháp thu thập thông tin**: Tìm hiểu tài liệu hướng dẫn chính thức từ Unity, tham khảo các khóa học online, tài liệu học thuật, và các game mẫu mã nguồn mở để học hỏi cách tổ chức code, thiết kế hệ thống.

**Phương pháp xử lý thông tin**: Tổng hợp, phân tích và lựa chọn các giải pháp lập trình phù hợp cho từng tính năng cụ thể. Sử dụng phương pháp định tính để đánh giá trải nghiệm người dùng qua quá trình thử nghiệm nội bộ.

**Phương pháp thực nghiệm**: Triển khai trực tiếp dự án trên Unity, thử nghiệm các kịch bản chơi khác nhau, ghi nhận lỗi và tối ưu hiệu suất dựa trên phản hồi và kết quả chạy thực tế.

.**1.6. Bố cục đề tài**

**Chương 2** trình bày cơ sở lý thuyết và các công nghệ liên quan đến quá trình phát triển game trong Unity, bao gồm kiến trúc Unity, thành phần cơ bản trong game, và một số mẫu thiết kế phổ biến được ứng dụng trong dự án.

**Chương 3** mô tả quá trình thiết kế và phát triển trò chơi, từ việc xác định yêu cầu, phân tích chức năng, xây dựng giao diện người dùng đến lập trình hành vi nhân vật, xử lý va chạm và lưu trữ dữ liệu.

**Chương 4** Kết luận

**CHƯƠNG 2. CƠ SỞ LÝ THUYẾT**

Sinh viên cần phải viết lời dẫn lý do chọn cơ sở lý thuyết áp dụng cho đề tài. Sau đó mới trình bày từng cơ sở lý thuyết được chọn (phải nêu được lịch sử hình thành và phát triển, khái niệm (định nghĩa), hình ảnh minh họa (nếu có), điểm mạnh, điểm yếu, v.v…)

**2.1. Unity Engine (ví dụ: 2.1. Unity Engine)**

Unity là một **nền tảng phát triển game đa nền tảng**, ra đời năm 2005 bởi Unity Technologies. Unity cho phép xây dựng game 2D, 3D trên nhiều thiết bị như PC, Android, iOS.

* **Ưu điểm**: giao diện trực quan, hỗ trợ vật lý, hoạt ảnh, UI, cộng đồng lớn.
* **Nhược điểm**: khó tối ưu nếu game phức tạp, dung lượng build lớn.

Unity giúp đơn giản hóa quá trình thiết kế scene, lập trình hành vi, và xuất bản sản phẩm.

**2.2. Ngôn ngữ lập trình C#**

C# là ngôn ngữ lập trình chính trong Unity. Được Microsoft phát triển từ năm 2000, C# có cú pháp rõ ràng, hỗ trợ hướng đối tượng, dễ học và tích hợp chặt với Unity.

* **Ưu điểm**: dễ viết, dễ bảo trì, mạnh mẽ trong xử lý logic.
* **Nhược điểm**: không tối ưu cao cho xử lý cấp thấp so với C++.

Trong game, C# dùng để lập trình điều khiển nhân vật, va chạm, điểm số, và logic gameplay.

**2.3. Lưu lại bản Save sử dụng .JSON**

JSON là định dạng văn bản nhẹ, lưu dữ liệu dạng key-value. Được dùng phổ biến để lưu cấu hình hoặc trạng thái người chơi.

* **Ưu điểm**: dễ đọc, dễ xử lý, linh hoạt.
* **Nhược điểm**: không bảo mật, không phù hợp cho dữ liệu lớn.

Trong đề tài, JSON được dùng để **lưu trạng thái người chơi** như level, điểm số và các tùy chọn.

**CHƯƠNG 3. PHÂN TÍCH VÀ XÂY DỰNG SẢN PHẨM GAME**

**3.1. Tổng quan trò chơi**

**3.1.2. Giới thiệu trò chơi**

• **GameName** –Leveling

• **Game Identity / Mantra**- Về 1 nhân vật farm Aura

. (Example: Stylized action plat former about a meatball fighting the dinner table.)

• **Design Pillars**- Farm.Aura.AuraFarming

• **Tag line**– “NOCHECKPOINT”

• **Art style**- –2D Pixel Game Once you’ve explained how to play your game, it’s important to show what your game will look like and what kind of art is behind it, since that will influence how the elements of your game’s universe will coexist, affecting the emotions of the players. This is a crucial point in the game’s marketing because it shows the appearance of the game and the feelings it will pass to the player.

**3.1.3. Gameplay**

Kinh nghiệm ở trong trò chơi có màu xanh gọi là Aura, thu thập để nâng cấp sức mạnh và trao đổi

**3.1.4. Cơ chế của trò chơi**

(Key Section)- These are topics that should be covered if relevant. All games are different and not everything applies. Consider this a checklist to help you capture all that is important. It should be complete enough that someone could build a clone of your game based on this detail.

• Rules– Chết chơi lại từ đầu

• Model of the game universe. Think of it as a simulation of a world, how do all the pieces interact? If it is a puzzle game, there may be no universe to discuss, but there is the layout and the structure of the puzzle. Included in the universe

– Physics– How does the physical universe work? If there are no physics, do not try to create them.– Economy

– What is the economy of the game?- Aura How does it work?- Đánh bại quái vật

• Character actions. These include

– Character movement– How does the character move?- Trái, phải, tăng tốc , leo tường

– Objects– how to pick them up and move them- E to interact. Are there key objects of importance that should be identified here? Is there an inventory that the player maintains or are all interactions transitory?

– Actions– What else can the player (or character) do? Include any in game switches and buttons that are used, interacting with objects, and what means of communication are used

– Combat– Ifthere is combat or even conflict, how is this specifically mod eled? What are the differences in weapons?

• Screen Flow– How each screen is related to every other and a description of the purpose of each screen

• Game Options- What are the options and how do they affect game play?-Trong Setting Sections và Save Replaying and saving Cheats and Easter Eggs if there are any

• Feedback for the Player– How can the player tell if they are doing well? Is there mechanisms to tell them what they are doing wrong

**3.1.5. Điều khiển**

E-Nhặt đồ/Tương tác với các Object khác -F-Stating which buttons/keys do what can be troublesome in the case where a single button performs more than one action (e.g. the ‘A’ button in any 3D Zelda). Start by including a simple picture of a controller or a keyboard with each button highlighted with their function in a more general sense. After that, if your game has advanced combos or something similar to that, explain them carefully, stating under which conditions each combo is “activated”.

**3.1.6. Cốt truyện**

(optional) If you have a puzzle game, there is probably no backstory or story. As soon as you add an avatar of any sort, you have a story. It may be as simple as who the character is.

• Backstory. If there is no plot or narrative in the story but you need to under stand the background, that is a backstory. The backstory can also be setting the stage for the story that is going to unfold.

• Plot elements. There are multiple patterns of a plot: the 3-act play, the hero’s journey, episodic, ... Use one of the models to describe the story of the game.

• Gamestory progression. Explain how the progression of the plot is exposed to the player. Remember that there are different approaches to the narrative: how it is exposed, who is telling it and the sequence that things happen. If you are using cut scenes to progress the story, describe the cut scenes in detail.

**3.1.7. Thế giới game**

(optional) Describe the general look and feel of world. Of particular importance is whether the world is broken up into different areas that are relevant to the game. If it is, describe how they are connected, how the player moves between them, and how they relate to levels - Dungeon

**3.1.8. Nhân vật và kẻ địch**

Player avatar. Indicate if the user has the ability to define their own avatar and what options they are able to define. List of characters in the game: identify which are playable and which are not. If there are characters that are relevant to the game but never seen, be sure to include them. Include enemies and opponents. If the ene mies are classes only, that is fine. Identify the class and if there is a specific number of them, etc. For each character, give their backstory, personality, appearance and abilities. Explain the relevance of each player to the story and their relationship to other characters. If certain of the characters only appear in certain levels or areas, explain that as well. This is also the place to describe any AI that is used for these characters.

**3.1.9. Màn chơi**

General description of how levels are used. If they are very similar, the detailed level descriptions need not be very long. For each level, describe what additional skills or information they are learning? For each level, give a synopsis of the level and the objectives of that level. If there is introductory material that is required, how is it provided? There are two key aspects to cover in each level: the game play changes and the content changes. In some cases, both change; in other cases, only one does. For example, as levels progress, you may give the player less hints so that they learn the content better and you may increase the frequency of enemies. For each level, describe how the map changes and whether it is a crafted or proce durally generated map. If it is procedurally generated, what are the constraints or requirements that put on it? Is there a training level or a playable tutorial? How is the player’s progress assessed at the end of the level? Are they allowed to replay levels? Does progression through the levels require a specific level of accomplish ment (that is are subsequent levels being unlocked)? Now let’s pretend we have a typical 2D platformer. One of the core elements of the game is the stages the player has to go through. It’s important that each stage feels unique so the player won’t feel as if they’re just repeating the same thing over and over again. On the other hand, the player should still be familiar with the flow of the stage, e.g. if there’s always a checkpoint somewhere halfway through it, or some collectible items along the way. What are the different types of enemies, terrains, doodads, and power-ups, and do they allow the level designers to come up with many different stages? You could present some beta stage diagrams to illustrate how will they be carried out.

**3.1.10. UI**

What is the visual system? Specifically, If there is a HUD, what is on it, how is it displayed and how is it accessed? What menus are included and how are they navigated? What is the camera model? Control System– How does the game player control the game? What are the specific commands? Audio, music, sound effects Game Art– intended style Help System Settings– what can the user change? Feedback for the Player– How can the player tell if they are doing well? Is there mechanisms to tell them what they are doing wrong? The head-up-display (HUD) is the in-game interface the player will have when playing the game. Rather than in-game menus like settings or inventory screens, this refers specifically to the floating windows and bars which don’t normally in teract with the game and serve an information-only purpose. This includes health bars, mini-maps, time counters, equipped items and their amounts, money, etc. Al though the size of the HUD will vary according to the game type (MMORPGs and RTSs will have big HUDs, while side-scrollers and puzzles will have very small ones), keep in mind that an HUD shouldn’t occupy too much of the screen.

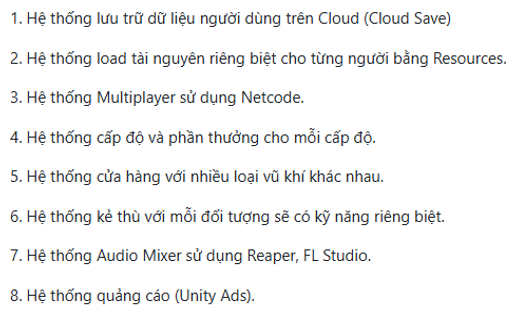
**3.2. Phân tích hệ thống game**

**Dựa trên lý do chọn đề tài và mục tiêu đề tài đạt được (đã được trình bày ở chương 1), xác định rõ ràng hơn về các yêu cầu cần thực hiện của đề tài được chọn. Tùy theo mỗi đề tài sinh viên thực hiện theo hướng dẫn của giảng viên hướng dẫn đề tài.**

**3.2.1. Các tác nhân của hệ thống**

**3.2.2. Chức năng hệ thống**

Ví dụ: (Phải trình bày chi tiết)



**3.2.3. Biểu đồ use-case**

**3.2.3. Biểu đồ lớp (Class Diagram)**

**3.2.3. Biểu đồ hoạt động (Activity Diagram)**

**3.2.4. Biểu đồ tuần tự (Sequence Diagram)**

**3.2.5. Biểu đồ trạng thái (State Diagram**)

**3.3. Xây dựng các phần thành của Game**

Phần này trình bày sản phẩm xây dựng được cho các thành phần của game trong hệ thống từ mục 3.1 và 3.2 đã phân tích.

**3.4. Xây dựng giao diện sản phẩm Game**

**(Phần Xây dựng giao diện cho các chức năng trong hệ thống tùy theo mỗi đề tài sẽ mà phân ra từng mục nhỏ khác nhau theo tác nhân hệ thống)**

**Lưu ý cách trình bày trong phần trình bày thiết kế giao diện:**

**Phải có tên chức năng của giao diện**

**Chức năng giúp người dùng thực hiện công việc gì, giải thích chức năng giao diện**

**Đặt tên hình cho chức năng, để làm danh mục hình, đặt đúng cứu pháp theo quy định.**

**KẾT LUẬN**

## **1. Kết luận về kết quả đạt được**

Sau quá trình tìm hiểu, nghiên cứu và triển khai đề tài “Xây dựng trò chơi [Tên game] sử dụng Unity”, em đã đạt được những kết quả nhất định, phần nào đáp ứng được các mục tiêu đã đề ra ban đầu.

Cụ thể, em đã:

* Tìm hiểu và vận dụng được nền tảng Unity để thiết kế và xây dựng một trò chơi hoàn chỉnh theo thể loại [Platformer/Arcade/...], từ hệ thống nhân vật, môi trường đến các chức năng cơ bản như điều khiển, tương tác vật lý, tính điểm và giao diện người dùng (UI).
* Áp dụng kiến thức về lập trình hướng đối tượng với ngôn ngữ C#, tổ chức mã nguồn theo cấu trúc rõ ràng và sử dụng các thành phần cốt lõi trong Unity như GameObject, Rigidbody2D, Collider, Animation, Prefab và Scene.
* Thực hiện các thao tác lưu trữ dữ liệu đơn giản bằng PlayerPrefs, xử lý tương tác bàn phím và tạo giao diện thân thiện cho người chơi.
* Hiểu và tuân thủ quy trình phát triển phần mềm: từ phân tích yêu cầu, thiết kế, lập trình, đến kiểm thử và hoàn thiện sản phẩm.

Tuy nhiên, do giới hạn về thời gian và kinh nghiệm thực tế, một số tính năng nâng cao như hệ thống AI cho kẻ địch, hệ thống vật phẩm (Inventory), lưu game phức tạp, hoặc khả năng xuất bản game đa nền tảng chưa được triển khai. Dù vậy, qua đề tài này, em đã tích lũy được những kiến thức thực tiễn quý giá trong lĩnh vực phát triển trò chơi, cũng như nâng cao tư duy logic, kỹ năng làm việc độc lập và kỹ năng giải quyết vấn đề.

## **2. Hướng phát triển trong tương lai**

Trên cơ sở kết quả đạt được và các chức năng còn hạn chế, em định hướng phát triển trò chơi theo các hướng sau:

* **Hoàn thiện các chức năng chưa làm được**:
  + Bổ sung hệ thống lưu – tải game bằng JSON hoặc ScriptableObject để đảm bảo khả năng mở rộng.
  + Tối ưu hiệu suất hoạt động khi chạy trên thiết bị cấu hình thấp.
  + Tinh chỉnh các yếu tố vật lý (va chạm, tốc độ, trọng lực) để tạo cảm giác chơi chân thực hơn.
* **Nâng cấp, cải tiến và mở rộng game**:
  + Phát triển thêm nhiều màn chơi (level) với độ khó tăng dần.
  + Thêm hệ thống vật phẩm, kỹ năng hoặc điểm thưởng để tăng tính tương tác.
  + Tích hợp âm thanh, hiệu ứng hình ảnh và animation để tăng trải nghiệm người chơi.
  + Hỗ trợ xuất bản game lên các nền tảng khác như Android, WebGL hoặc iOS.
  + Áp dụng kiến thức về thiết kế UI/UX để cải tiến giao diện game thân thiện, hiện đại hơn.
  + Tìm hiểu và tích hợp các công nghệ như Multiplayer (Photon) hoặc AI để nâng cao chất lượng gameplay.

Với nền tảng kiến thức đã tích lũy cùng sản phẩm ban đầu, em tin rằng đề tài có tiềm năng phát triển xa hơn nữa trong tương lai, không chỉ phục vụ cho mục tiêu học tập mà còn có thể ứng dụng vào thực tế trong lĩnh vực công nghiệp trò chơi điện tử đang phát triển mạnh mẽ hiện nay.

**DANH MỤC TÀI LIỆU THAM KHẢO**

(Tạo tài liệu tham khảo tự động, xem hướng dẫn)

Tài liệu tham khảo (bao gồm cả tên Website): sử dụng hình thức trích dẫn tài liệu tham khảo phổ biến sau đây:

- IEEE (Institute of Electrical and Electronics Engineers): thường áp dụng cho lĩnh vực kĩ thuật, khoa học máy tính, công nghệ thông tin.

**Ví dụ:**

[1] Ui-Jeong Lee, Sang-Jun Ahn , Dong-Young Choi, Sang-Min Chin and Dae-Sung Jang (2023) Airspace Designs and Operations for UAS Traffic Management at Low Altitude, Aerospace, 10(9):737. DOI: 10.3390/aerospace10090737

[2] Pedro Maristany de las Casas, Antonio Sedeño-Noda, Ralf Borndörfer (2021) An Improved Multiobjective Shortest Path Algorithm, Computers & Operations Research, Volume 135, pp. 424-449. DOI: 10.1016/j.cor.2021.105424

**PHỤ LỤC**

Đưa các Link sau:

* Link kế hoạch làm việc của nhóm, phân chia công việc hằng tuần.
* Link GitHub. [Duckkeip/LapTrinhGameN4Final](https://github.com/Duckkeip/LapTrinhGameN4Final)