TRƯỜNG ĐẠI HỌC TRÀ VINH

KHOA KỸ THUẬT VÀ CÔNG NGHỆ

**BỘ MÔN CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO MÔN HỌC**

**ỨNG DỤNG QUẢN LÝ LỊCH TRÌNH CÁ NHÂN**

|  |  |
| --- | --- |
| **Giáo viên giảng dạy:**  TS. Nguyễn Bảo Ân | **Sinh viên thực hiện:**  110122193 Thạch Thị Huệ Trinh DA22TTC  110122196 Lê Xuân Trường DA22TTC |

***Trà Vinh, tháng 7 năm 2025***

TRƯỜNG ĐẠI HỌC TRÀ VINH

KHOA KỸ THUẬT VÀ CÔNG NGHỆ

**BỘ MÔN CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO MÔN HỌC**

**ỨNG DỤNG QUẢN LÝ LỊCH TRÌNH CÁ NHÂN**

|  |  |
| --- | --- |
| **Giáo viên giảng dạy:**  TS. Nguyễn Bảo Ân | **Sinh viên thực hiện:**  110122193 Thạch Thị Huệ Trinh DA22TTC  110122196 Lê Xuân Trường DA22TTC |

***Trà Vinh, tháng 7 năm 2025***

**NHẬN XÉT CỦA GIÁO VIÊN**

*Trà Vinh, ngày ….. tháng …… năm ……*

**Giáo viên giảng dạy**

*(Ký tên và ghi rõ họ tên)*

# MỤC LỤC

[MỤC LỤC 1](#_Toc202539798)

[DANH MỤC HÌNH ẢNH 3](#_Toc202539799)

[CHƯƠNG 1. GIỚI THIỆU 4](#_Toc202539800)

[1.1. Tên đề tài và chủ đề 4](#_Toc202539801)

[1.2. Mục tiêu của ứng dụng 5](#_Toc202539802)

[1.3. Lí do chọn đề tài 5](#_Toc202539803)

[1.4. Các đối tượng trong trò chơi 6](#_Toc202539804)

[1.4.1. Người chơi (Player) 6](#_Toc202539805)

[1.4.2. Mảnh ghép hình ảnh (Images) 6](#_Toc202539806)

[1.4.3. Ô trống (Blank Tile) 6](#_Toc202539807)

[1.4.4. Thuật toán A\* 6](#_Toc202539808)

[CHƯƠNG 2. CƠ SỞ LÝ THUYẾT 8](#_Toc202539809)

[2.1. Giới thiệu về trí tuệ nhân tạo 8](#_Toc202539810)

[2.2. Giới thiệu về Unity 9](#_Toc202539811)

[2.3. Giới thiệu về ngôn ngữ lập trình C# 10](#_Toc202539812)

[2.4. Giới thiệu về thuật toán A-Star 10](#_Toc202539813)

[2.5. Giới thiệu trò chơi xếp hình 11](#_Toc202539814)

[2.6. Quy luật trò chơi 13](#_Toc202539815)

[CHƯƠNG 3. KẾT QUẢ THỰC HIỆN 14](#_Toc202539816)

[3.1. Một số hình ảnh trong trò chơi 14](#_Toc202539817)

[3.1.1. Giao diện ban đầu 14](#_Toc202539818)

[3.1.2. Giao diện sau khi xáo trộn 15](#_Toc202539819)

[3.1.3. Giao diện kết thúc 16](#_Toc202539820)

[3.1.4. Giao diện giải bằng A Star 16](#_Toc202539821)

[3.2. Trạng thái kết thúc của trò chơi 17](#_Toc202539822)

[3.3. Một số kinh nghiệm trong trò chơi 19](#_Toc202539823)

[3.3.1. Ưu tiên giải quyết theo từng hàng hoặc cột 19](#_Toc202539824)

[3.3.2. Hạn chế làm rối phần đã hoàn thành 19](#_Toc202539825)

[3.3.3. Sử dụng ô trống thông minh 20](#_Toc202539826)

[3.3.4. Kiên nhẫn và đánh giá kỹ bước đi 20](#_Toc202539827)

[3.3.5. Tận dụng thuật toán A Star 20](#_Toc202539828)

[CHƯƠNG 4. ĐÁNH GIÁ VÀ HƯỚNG PHÁT TRIỂN 22](#_Toc202539829)

[4.1. Đánh giá 22](#_Toc202539830)

[4.1.1. Ưu điểm của trò chơi 22](#_Toc202539831)

[4.1.2. Hạn chế của trò chơi 22](#_Toc202539832)

[4.2. Hướng phát triển 23](#_Toc202539833)

[CHƯƠNG 5. KẾT LUẬN 24](#_Toc202539834)

[DANH MỤC TÀI LIỆU THAM KHẢO 26](#_Toc202539835)

# DANH MỤC HÌNH ẢNH

[Hình 2.1 Trí tuệ nhân tạo (Artificial Intelligence – AI) là ngành khoa học [1] 7](#_Toc198749270)

[Hình 2.2 Unity là một nền tảng phát triển game đa nền tảng [2] 8](#_Toc198749271)

[Hình 2.3 Thuật toán A\* là một thuật toán tìm kiếm [3] 10](#_Toc198749272)

[Hình 3.1 Hình ảnh giao diện ban đầu của trò chơi 13](#_Toc198749273)

[Hình 3.2 Hình ảnh giao diện sau khi xáo trộn 14](#_Toc198749274)

[Hình 3.3 Hình ảnh giao kết thúc của trò chơi 15](#_Toc198749275)

[Hình 3.4 Hình ảnh giao diện giải bằng A Star 16](#_Toc198749276)

[Hình 3.5 Hình ảnh giao diện chiến thắng trong trò chơi 17](#_Toc198749277)

# GIỚI THIỆU

## Tên đề tài và chủ đề

Tên đề tài: Ứng dụng quản lý lịch trình cá nhân

Chủ đề nghiên cứu: Phát triển một ứng dụng web nhằm hỗ trợ người dùng trong việc quản lý và theo dõi lịch trình cá nhân với các chế độ hiển thị linh hoạt theo ngày, tuần và tháng. Hệ thống cho phép thực hiện các thao tác nghiệp vụ cơ bản như tạo mới, chỉnh sửa, xoá sự kiện, đồng thời đảm bảo tính nhất quán và liên tục của dữ liệu thông qua cơ chế đồng bộ với máy chủ thông qua giao tiếp API.

Trong quá trình xây dựng hệ thống, đề tài ứng dụng các công nghệ và công cụ tiêu biểu trong lĩnh vực công nghệ phần mềm hiện nay, cụ thể như sau:

* Thiết kế giao diện người dùng (Frontend): Xây dựng giao diện trực quan, tối ưu trải nghiệm người dùng, hỗ trợ tương tác lịch trình thông qua trình duyệt web, sử dụng các công nghệ như HTML, CSS và JavaScript.
* Phát triển hệ thống API theo kiến trúc RESTful (Backend): Thiết kế và triển khai các điểm cuối (endpoint) phục vụ cho việc lưu trữ, truy xuất và xử lý dữ liệu lịch trình một cách hiệu quả.
* Quản lý mã nguồn với GitHub: Đảm bảo kiểm soát phiên bản, đồng bộ hóa tiến độ phát triển và hỗ trợ làm việc nhóm trong môi trường phân tán.

Ứng dụng các công cụ hỗ trợ phát triển phần mềm, bao gồm:

* Jira: Quản lý tiến độ và công việc theo mô hình Agile, hỗ trợ phân chia nhiệm vụ và theo dõi tiến trình phát triển.
* Docker: Tạo lập môi trường phát triển và triển khai nhất quán, đảm bảo tính đồng bộ giữa các môi trường hệ thống.
* Postman: Hỗ trợ kiểm thử và mô phỏng các truy vấn đến API một cách trực quan và hiệu quả.
* Swagger: Tài liệu hóa API và cung cấp giao diện kiểm thử tự động, nâng cao khả năng kiểm tra và bảo trì hệ thống.

## Mục tiêu của ứng dụng

Mục tiêu chính của đề tài là xây dựng một hệ thống ứng dụng web cho phép người dùng quản lý lịch trình cá nhân một cách hiệu quả, trực quan và có khả năng mở rộng. Ứng dụng đáp ứng các chức năng cốt lõi bao gồm:

* Hiển thị lịch trình theo chế độ ngày, tuần, tháng, hỗ trợ quan sát tổng thể các sự kiện đã được thiết lập.
* Cung cấp khả năng tạo mới, chỉnh sửa, xoá sự kiện tương ứng với từng ngày cụ thể trên giao diện lịch.
* Hiển thị chi tiết nội dung của các sự kiện khi người dùng tương tác với từng ô ngày.
* Tự động cập nhật và hiển thị lại giao diện khi có sự thay đổi dữ liệu.
* Đồng bộ hoá dữ liệu với máy chủ thông qua hệ thống API, đảm bảo tính nhất quán, an toàn và khả năng lưu trữ lâu dài.

## Lí do chọn đề tài

Quản lý lịch trình là một nhu cầu phổ biến trong nhiều lĩnh vực như giáo dục, hành chính, doanh nghiệp và đời sống cá nhân. Việc ghi nhớ và sắp xếp thời gian một cách hợp lý đóng vai trò quan trọng trong việc nâng cao hiệu suất làm việc và tổ chức công việc khoa học. Tuy nhiên, phần lớn các giải pháp hiện tại thường tích hợp trong các hệ sinh thái phức tạp, yêu cầu tài khoản đồng bộ, hoặc không phù hợp với các tình huống sử dụng độc lập, đơn giản.

Với mục tiêu xây dựng một ứng dụng có giao diện trực quan, chức năng tập trung, dễ sử dụng và dễ triển khai, đề tài hướng đến việc giải quyết một cách cụ thể bài toán quản lý lịch trình cá nhân theo cách tiếp cận nhẹ, hiệu quả và khả thi.

Đồng thời, việc lựa chọn đề tài còn nhằm hiện thực hóa một hệ thống phần mềm hoàn chỉnh theo đúng các giai đoạn của quy trình phát triển phần mềm hiện đại, từ đặc tả yêu cầu, thiết kế, lập trình đến kiểm thử và triển khai. Đây là một trong những hướng tiếp cận phù hợp để đánh giá khả năng áp dụng lý thuyết vào thực tiễn cũng như kiểm chứng các nguyên tắc thiết kế phần mềm thông qua sản phẩm cụ thể.

## Các đối tượng trong trò chơi

### Người chơi (Player)

Người chơi là đối tượng chính tương tác trực tiếp với trò chơi thông qua giao diện người dùng. Người chơi sử dụng chuột để di chuyển các mảnh ghép vào ô trống, từng bước sắp xếp lại các mảnh theo đúng trật tự. Mục tiêu của người chơi là đưa bảng xếp hình từ trạng thái ban đầu (xáo trộn ngẫu nhiên) về trạng thái hoàn chỉnh trong thời gian ngắn nhất hoặc số bước di chuyển ít nhất. Ngoài ra, người chơi có thể sử dụng nút “Giải bằng Astar” để xem gợi ý lời giải dựa trên thuật toán A\*.

### Mảnh ghép hình ảnh (Images)

Mỗi mảnh ghép là một ô hình vuông chứa một phần của bức ảnh lớn được chia nhỏ thành 8 phần bằng nhau. Các mảnh này được sắp xếp trên một lưới 3x3, trong đó có một ô trống để tạo điều kiện cho việc di chuyển các mảnh ghép. Người chơi có thể di chuyển một mảnh hình ảnh sang vị trí liền kề với ô trống theo chiều ngang hoặc dọc. Mục tiêu của trò chơi là sắp xếp lại các mảnh ghép sao cho bức ảnh được ghép lại hoàn chỉnh như ban đầu. Mỗi mảnh có vị trí và tọa độ riêng để hệ thống theo dõi, xử lý các thao tác di chuyển cũng như kiểm tra trạng thái kết thúc của trò chơi.

### Ô trống (Blank Tile)

Ô trống là một vị trí không có mảnh ghép, được tạo ra bằng cách loại bỏ một mảnh bất kỳ khỏi lưới 3x3 (vị trí thứ 9). Đây là vị trí cho phép người chơi thực hiện các thao tác hoán đổi mảnh. Các mảnh ghép chỉ có thể được di chuyển nếu chúng nằm liền kề ô trống theo chiều ngang hoặc dọc. Ô trống cũng được hệ thống sử dụng để xác định các bước đi hợp lệ khi áp dụng thuật toán A\*.

### Thuật toán A\*

Thuật toán A\* được tích hợp vào phần logic ẩn của trò chơi, đóng vai trò như “bộ não” để tìm ra lộ trình tối ưu từ trạng thái xáo trộn đến trạng thái hoàn chỉnh. Thuật toán hoạt động dựa trên nguyên tắc tìm kiếm theo tổng chi phí f(n) = g(n) + h(n), trong đó g(n) là chi phí từ trạng thái ban đầu đến trạng thái hiện tại, còn h(n) là ước lượng khoảng cách còn lại đến đích. Thuật toán này được sử dụng để tự động giải trò chơi.

# CƠ SỞ LÝ THUYẾT

## Giới thiệu về trí tuệ nhân tạo

Trí tuệ nhân tạo (Artificial Intelligence – AI) là ngành khoa học và công nghệ nghiên cứu và phát triển các hệ thống máy tính có khả năng thực hiện các nhiệm vụ mà con người thường cần đến trí tuệ để giải quyết, như nhận thức, học tập, suy luận, lập kế hoạch và tự điều chỉnh hành vi. AI bao gồm nhiều lĩnh vực con như học máy (machine learning), xử lý ngôn ngữ tự nhiên (natural language processing), thị giác máy tính (computer vision) và các thuật toán tìm kiếm. Trong đó, các thuật toán tìm kiếm đóng vai trò quan trọng trong việc giải quyết các bài toán phức tạp, tối ưu hóa đường đi hoặc trạng thái, đặc biệt trong các ứng dụng về game và robot.

Việc ứng dụng AI trong phát triển trò chơi không chỉ nâng cao trải nghiệm người chơi mà còn giúp mô phỏng các hành vi thông minh, hỗ trợ người chơi hoặc tự động giải quyết các thử thách trong game. Thuật toán A\* (A-star) là một trong những thuật toán tìm kiếm nổi bật, được sử dụng phổ biến trong các bài toán tìm đường đi ngắn nhất, tối ưu hóa trạng thái trong các trò chơi và ứng dụng thực tế khác.[1]



Hình 2.1 Trí tuệ nhân tạo (Artificial Intelligence – AI) là ngành khoa học [1]

## Giới thiệu về Unity

Unity là một nền tảng phát triển game đa nền tảng, được sử dụng rộng rãi trong ngành công nghiệp game cũng như các ứng dụng tương tác khác như mô phỏng, thực tế ảo (VR) và thực tế tăng cường (AR). Được phát triển bởi Unity Technologies, Unity cung cấp một môi trường phát triển tích hợp (IDE) mạnh mẽ với giao diện đồ họa trực quan, giúp các nhà phát triển dễ dàng thiết kế, xây dựng và triển khai các trò chơi 2D và 3D.

Unity hỗ trợ nhiều ngôn ngữ lập trình, trong đó phổ biến nhất là C#, cho phép các lập trình viên viết mã điều khiển các hành vi và logic của trò chơi một cách linh hoạt và hiệu quả. Với khả năng chạy trên nhiều nền tảng như Windows, macOS, Android, iOS, WebGL, và nhiều thiết bị console, Unity giúp tiết kiệm thời gian và chi phí phát triển bằng cách cho phép tái sử dụng mã nguồn trên nhiều hệ điều hành khác nhau.

Ngoài ra, Unity còn có một cộng đồng người dùng lớn và kho tài nguyên phong phú (Asset Store) hỗ trợ phát triển nhanh chóng và đa dạng các tính năng cho trò chơi. Nhờ những ưu điểm này, Unity đã trở thành công cụ phổ biến trong việc phát triển các dự án game, từ những trò chơi nhỏ cho đến các sản phẩm game chuyên nghiệp. [2]



Hình 2.2 Unity là một nền tảng phát triển game đa nền tảng [2]

## Giới thiệu về ngôn ngữ lập trình C#

C# (phát âm là "C-Sharp") là một ngôn ngữ lập trình hiện đại, hướng đối tượng do Microsoft phát triển vào đầu những năm 2000, nằm trong nền tảng .NET. C# được thiết kế với mục tiêu cân bằng giữa tính mạnh mẽ, an toàn, dễ sử dụng và hiệu suất cao, phù hợp với nhiều mô hình phát triển phần mềm từ ứng dụng desktop, web, đến game và hệ thống phân tán. [4]

Một số đặc điểm nổi bật của C#:

* Hướng đối tượng toàn diện: Hỗ trợ đầy đủ các khái niệm như kế thừa, đóng gói, đa hình.
* Được quản lý bởi CLR (Common Language Runtime): Tăng tính an toàn bộ nhớ, kiểm soát luồng tốt hơn, quản lý bộ nhớ tự động (garbage collection).
* Hỗ trợ LINQ, async/await và lambda expressions: Giúp viết mã ngắn gọn, hiệu quả và dễ bảo trì.
* Tích hợp mạnh mẽ với Visual Studio: Một trong những IDE mạnh nhất hiện nay.
* Được sử dụng phổ biến trong phát triển game với Unity, lập trình web với ASP.NET, ứng dụng desktop với WinForms hoặc WPF.

C# hiện vẫn được duy trì và phát triển tích cực bởi Microsoft và cộng đồng .NET Foundation, với nhiều bản cập nhật mới hàng năm, mở rộng khả năng lập trình hiệu suất cao, đa nền tảng và hiện đại.

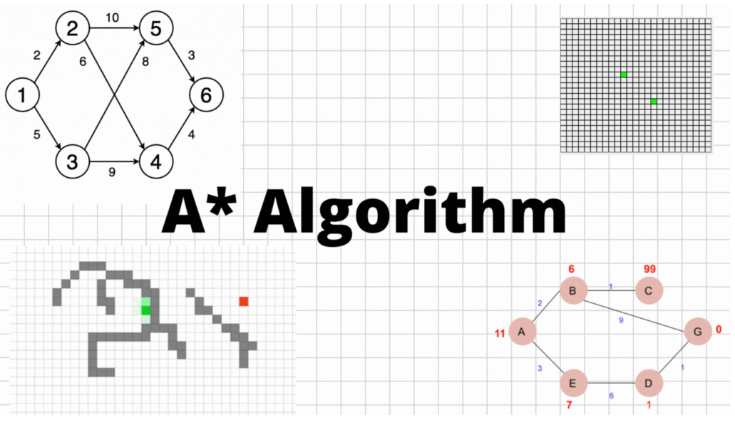
## Giới thiệu về thuật toán A-Star

Thuật toán A\* là một thuật toán tìm kiếm được sử dụng phổ biến trong lĩnh vực trí tuệ nhân tạo, đặc biệt là trong các bài toán tìm đường đi ngắn nhất và giải quyết trạng thái trong không gian tìm kiếm. A\* được giới thiệu lần đầu tiên vào năm 1968 bởi Peter Hart, Nils Nilsson và Bertram Raphael. Thuật toán này kết hợp giữa phương pháp tìm kiếm theo chi phí thực (như Dijkstra) và tìm kiếm có hướng dẫn bằng hàm heuristic (như Best-First Search), nhờ đó vừa đảm bảo tìm được lời giải tối ưu vừa tăng hiệu quả tìm kiếm.

A\* sử dụng một hàm đánh giá f(n) cho mỗi đỉnh n trong không gian tìm kiếm, được tính theo công thức: f(n) = g(n) + h(n). Trong đó:

* g(n) là chi phí thực từ điểm bắt đầu đến đỉnh hiện tại n.
* h(n) là giá trị ước lượng (heuristic) chi phí từ đỉnh n đến đích.

Thuật toán A\* sẽ chọn mở rộng các đỉnh có giá trị f(n) thấp nhất. Nếu hàm heuristic h(n) được thiết kế tốt (ví dụ: không vượt quá chi phí thực), A\* sẽ luôn đảm bảo tìm được đường đi tối ưu. [3]



Hình 2.3 Thuật toán A\* là một thuật toán tìm kiếm [3]

## Giới thiệu trò chơi xếp hình

Trò chơi xếp hình là một trò chơi dạng câu đố truyền thống, trong đó người chơi phải sắp xếp lại các mảnh ghép hình ảnh về đúng vị trí ban đầu để tạo thành một bức tranh hoàn chỉnh, thông qua các thao tác di chuyển hợp lệ. Trong phiên bản phổ biến nhất – trò chơi xếp hình 3x3 bao gồm 8 mảnh hình ảnh được cắt từ một bức tranh lớn và một ô trống, tất cả được đặt trên lưới kích thước 3x3. Mỗi lần chơi, các mảnh hình ảnh được xáo trộn ngẫu nhiên, và người chơi phải khôi phục lại bố cục ban đầu của bức tranh bằng cách di chuyển các mảnh hình ảnh vào vị trí ô trống.

Về mặt toán học và tin học, trò chơi xếp hình là một ví dụ điển hình của bài toán tìm kiếm trong không gian trạng thái hữu hạn, trong đó:

* Trạng thái (state): là một hoán vị cụ thể của các mảnh hình ảnh trên lưới 3x3.
* Hành động (action): là việc di chuyển một mảnh hình ảnh liền kề vào vị trí ô trống.
* Trạng thái mục tiêu (goal state): là cấu trúc hoàn chỉnh của hình ảnh gốc khi các mảnh ghép được đặt đúng vị trí.
* Không gian trạng thái (state space): bao gồm tất cả các trạng thái có thể đạt được từ trạng thái ban đầu thông qua các hành động hợp lệ.

Tổng cộng có 9! = 362,880 cách hoán vị các mảnh ghép và ô trống. Tuy nhiên, chỉ có một nửa trong số đó (tức 181,440 trạng thái) là có thể giải được, do ràng buộc về tính khả đảo (solvability) của bài toán. Một trạng thái chỉ có thể được giải thành công nếu số lượng hoán vị nghịch (inversion count) của nó thỏa mãn điều kiện xác định trước [1]

Trong lĩnh vực Trí tuệ nhân tạo (AI), trò chơi xếp hình được sử dụng như một mô hình điển hình để nghiên cứu và áp dụng các thuật toán tìm kiếm trạng thái. Một số thuật toán thường được dùng gồm:

* Tìm kiếm vét cạn (Brute-force search)
* Tìm kiếm theo chiều sâu (Depth-First Search)
* Tìm kiếm theo chiều rộng (Breadth-First Search)
* Tìm kiếm có hướng dẫn (Greedy Search)
* Tìm kiếm tối ưu A\* (A-star Search)

Các thuật toán này vận hành dựa trên hàm chi phí để mở rộng các trạng thái, từ đó tìm ra chuỗi hành động tối ưu nhằm đưa trò chơi từ trạng thái ban đầu về trạng thái hoàn chỉnh một cách hiệu quả nhất.

## Quy luật trò chơi

Trò chơi xếp hình 3x3 sử dụng các mảnh ghép là các phần hình ảnh được cắt nhỏ từ một bức tranh lớn. Bàn chơi được bố trí trên một lưới vuông 3x3, gồm 8 mảnh hình ảnh và một ô trống (blank tile). Mục tiêu của người chơi là sắp xếp lại các mảnh ghép sao cho tạo thành hình ảnh hoàn chỉnh thông qua các bước di chuyển hợp lệ.

Các quy tắc chính của trò chơi bao gồm:

* Di chuyển mảnh ghép vào ô trống: Người chơi chỉ được phép di chuyển một mảnh ghép nằm cạnh (trên, dưới, trái hoặc phải) ô trống vào vị trí ô trống đó.
* Chỉ di chuyển một mảnh mỗi lượt: Mỗi bước di chuyển chỉ thay đổi vị trí của duy nhất một mảnh ghép.
* Giữ nguyên phạm vi trong lưới: Mảnh ghép không thể di chuyển chéo hoặc ra ngoài phạm vi 3x3 của bàn chơi.
* Trạng thái kết thúc: Trò chơi kết thúc khi các mảnh ghép được sắp xếp đúng thứ tự, tạo thành hình ảnh gốc hoàn chỉnh, với ô trống nằm ở vị trí cuối cùng (góc dưới bên phải của lưới).
* Trạng thái khởi đầu hợp lệ: Trạng thái ban đầu của trò chơi là trạng thái hoàn chỉnh, tức là các mảnh ghép đã được sắp xếp đúng vị trí tạo thành hình ảnh nguyên vẹn. Khi người chơi nhấn nút "Xáo trộn", các mảnh ghép sẽ được tráo đổi vị trí một cách ngẫu nhiên nhưng vẫn đảm bảo tính khả thi để trò chơi có thể được giải. Điều này nghĩa là chỉ những cách xáo trộn mà từ đó người chơi có thể đưa các mảnh trở về trạng thái hoàn chỉnh mới được phép sử dụng làm trạng thái khởi đầu của mỗi ván chơi.

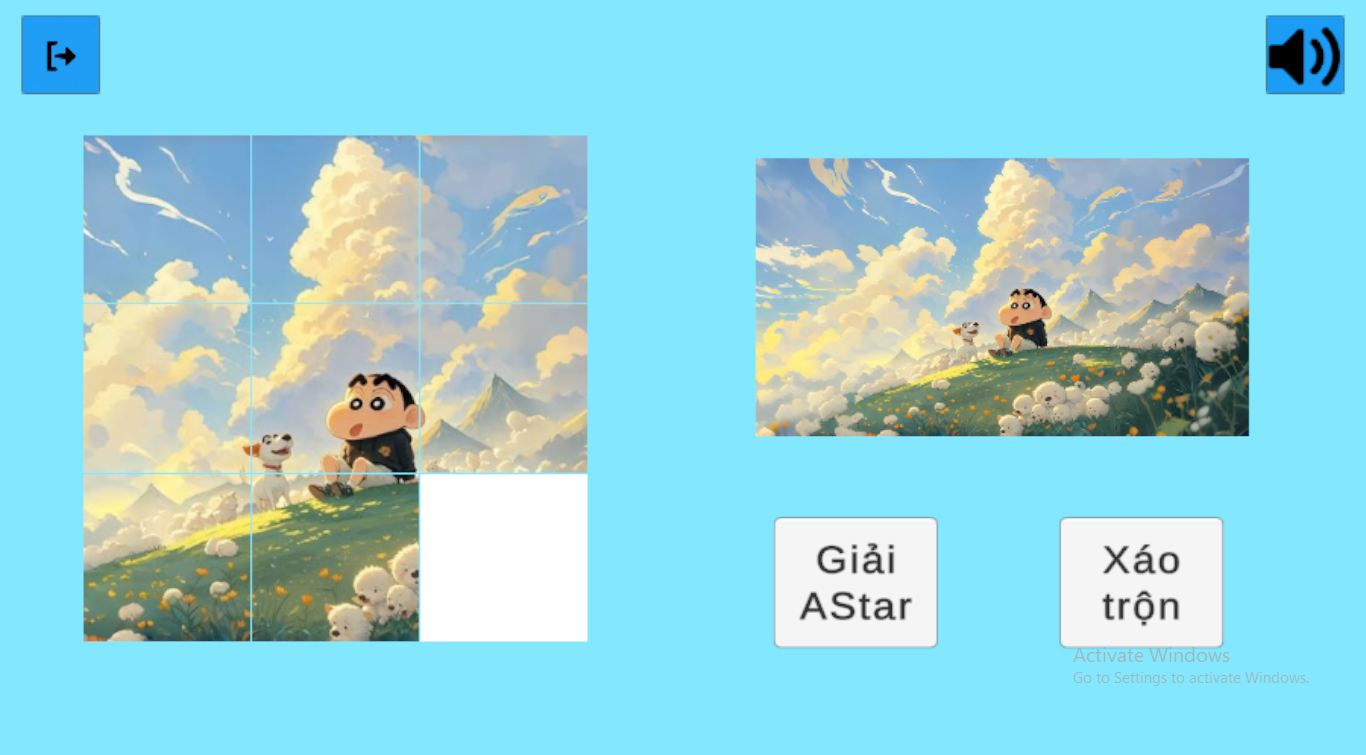
Những quy tắc này đảm bảo tính thách thức của trò chơi cũng như tính chặt chẽ trong quá trình áp dụng các thuật toán tìm kiếm như A\* để tự động giải hoặc hỗ trợ người chơi.

# KẾT QUẢ THỰC HIỆN

## Một số hình ảnh trong trò chơi

### Giao diện ban đầu

Giao diện khởi đầu của trò chơi hiển thị bức ảnh đầy đủ ở trạng thái hoàn chỉnh, được chia thành 9 ô vuông tạo thành lưới 3x3. Trong đó có 8 ô chứa các phần của hình ảnh, và 1 ô trống nằm ở vị trí cuối cùng (góc phải dưới cùng).



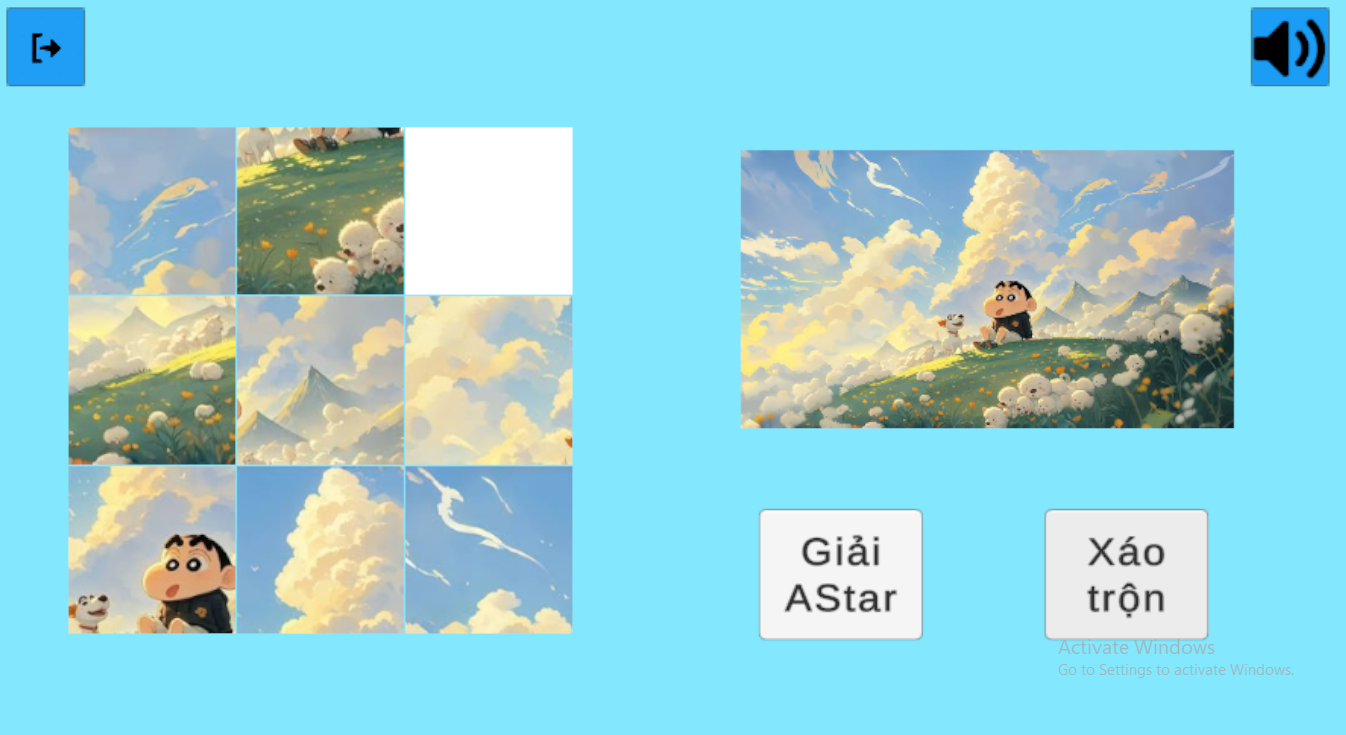
Hình 3.1 Hình ảnh giao diện ban đầu của trò chơi

Trên giao diện có các nút chức năng chính như:

* Nút “Xáo trộn”: Khi nhấn vào, hệ thống sẽ hoán đổi vị trí các ô ảnh một cách ngẫu nhiên nhưng đảm bảo tạo ra trạng thái có thể giải được.
* Nút “Giải bằng A Star”: Kích hoạt thuật toán A để tự động hiển thị các bước di chuyển từ trạng thái hiện tại đến trạng thái hoàn chỉnh, giúp người chơi quan sát cách giải logic của thuật toán.
* Nút “Âm thanh”: Khi được bật, nút này sẽ phát nhạc nền trong suốt quá trình chơi để tạo không khí sinh động và thư giãn. Người chơi cũng có thể nhấn lại để tắt/bật âm thanh tùy ý.
* Nút “Thoát”: Dùng để đóng trò chơi và thoát khỏi ứng dụng một cách an toàn.

### Giao diện sau khi xáo trộn

Sau khi người chơi nhấn nút “Xáo trộn”, hệ thống sẽ tự động hoán đổi vị trí các mảnh hình ảnh một cách ngẫu nhiên, tạo ra một cấu hình mới dùng làm trạng thái khởi đầu của trò chơi. Thuật toán đảm bảo rằng trạng thái sau xáo trộn là hợp lệ và có thể giải được, tức là có thể đưa về trạng thái hoàn chỉnh bằng một chuỗi bước di chuyển hợp lý.



Hình 3.2 Hình ảnh giao diện sau khi xáo trộn

Giao diện hiển thị một lưới 3x3 gồm:

* 8 mảnh hình ảnh được phân bố ngẫu nhiên trên lưới.
* 1 ô trống nằm ở một vị trí bất kỳ, được dùng làm không gian tạm để di chuyển các mảnh liền kề.

Người chơi sẽ sử dụng các thao tác nhấp chuột để di chuyển các mảnh hình vào ô trống nhằm đưa toàn bộ lưới trở về trạng thái ban đầu.

Trong quá trình chơi, các nút như *“***Giải bằng A Star***”* và **“Thoát”** vẫn khả dụng, cho phép người chơi chọn giải tự động hoặc thoát khỏi trò chơi khi cần.

### Giao diện kết thúc

Khi người chơi hoàn tất việc sắp xếp các mảnh ghép sao cho toàn bộ hình ảnh được khôi phục về đúng vị trí ban đầu, trò chơi sẽ nhận diện trạng thái hiện tại đã trùng khớp với trạng thái mục tiêu và hiển thị thông báo đã hoàn thành trò chơi.

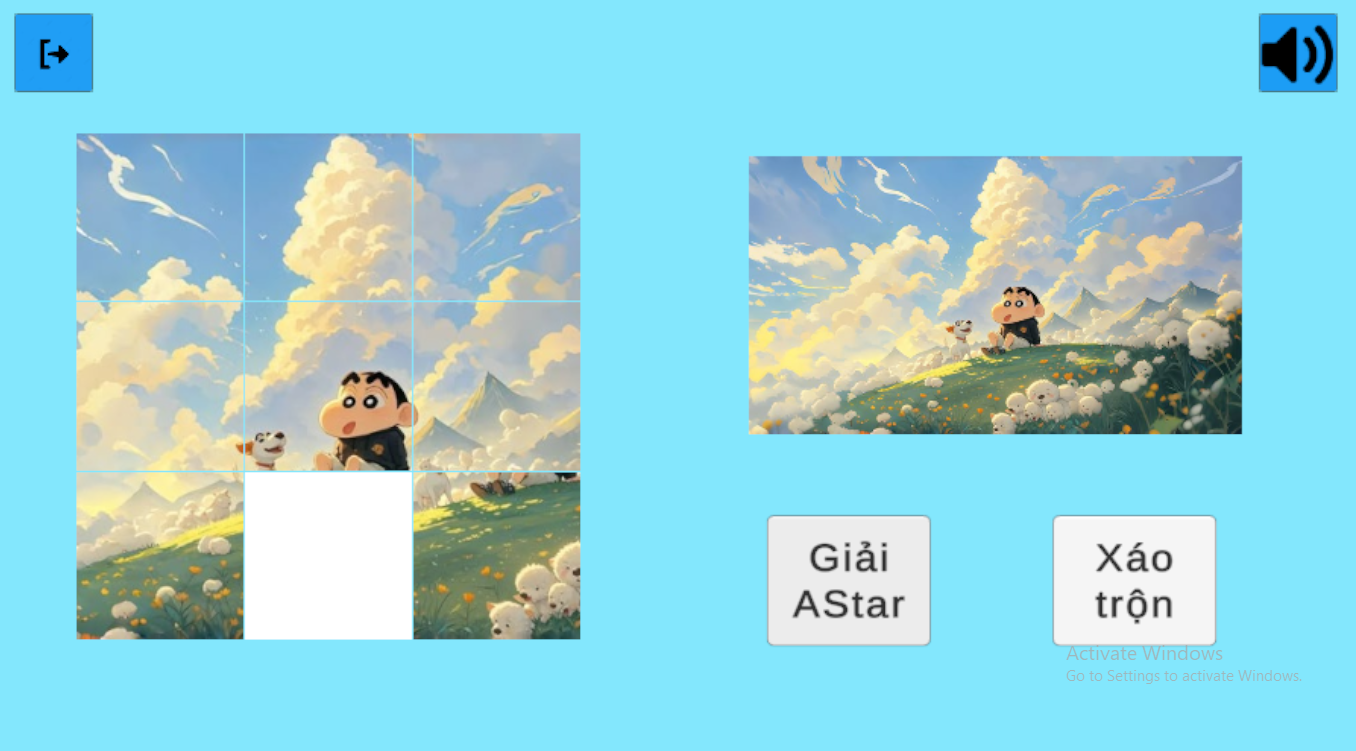


Hình 3.3 Hình ảnh giao kết thúc của trò chơi

* Toàn bộ 8 mảnh hình ảnh được đặt đúng vị trí từ trái sang phải, từ trên xuống dưới trên lưới 3x3.
* Ô trống nằm ở góc dưới cùng bên phải , đúng theo cấu hình chuẩn.

### Giao diện giải bằng A Star

Khi người chơi nhấn nút “Giải bằng A\*”, hệ thống sẽ kích hoạt thuật toán A\* để tìm ra chuỗi bước di chuyển ngắn nhất từ trạng thái hiện tại đến trạng thái hoàn chỉnh.



Hình 3.4 Hình ảnh giao diện giải bằng A Star

Giao diện lúc này có các đặc điểm như sau:

* Hiển thị từng bước di chuyển tự động: Các mảnh ghép sẽ được hệ thống tự động di chuyển lần lượt theo thứ tự mà thuật toán A\* tìm ra. Mỗi bước sẽ diễn ra trong khoảng thời gian ngắn để người chơi dễ theo dõi.
* Tạm thời khóa điều khiển thủ công: Trong quá trình A\* thực hiện giải, người chơi không thể tương tác thủ công với các mảnh ghép để tránh làm gián đoạn quá trình.
* Chuyển về trạng thái kết thúc: Sau khi hoàn tất, trò chơi trở về trạng thái hoàn chỉnh.

Giao diện giải tự động bằng A Star không chỉ giúp người chơi vượt qua thử thách, mà còn đóng vai trò như một công cụ trực quan hóa thuật toán A\*, từ đó hỗ trợ việc học và hiểu về trí tuệ nhân tạo.

## Trạng thái kết thúc của trò chơi

Trạng thái kết thúc là thời điểm trò chơi đạt đến mục tiêu cuối cùng, tức là tất cả các mảnh ghép đã được sắp xếp đúng vị trí trên lưới 3x3 để tái tạo lại hình ảnh ban đầu một cách hoàn chỉnh.

Cụ thể là:

* Tất cả 8 mảnh hình ảnh nằm đúng thứ tự, tạo thành một hình ảnh đầy đủ, liền mạch như khi chưa bị xáo trộn.
* Ô trống (Blank Tile) sẽ nằm ở góc dưới cùng bên phải của lưới, không ảnh hưởng đến tính toàn vẹn của hình ảnh.
* Không còn mảnh nào sai vị trí, hệ thống sẽ kiểm tra và xác nhận đây là trạng thái thắng cuộc (trạng thái kết thúc).

Khi trạng thái kết thúc được xác định giao diện sẽ tự động hiển thị thông báo “Chúc mừng! Bạn đã hoàn thành trò chơi!” để thông báo cho người chơi biết rằng họ đã chiến thắng.

Hệ thống sẽ vô hiệu hóa các thao tác di chuyển tiếp theo để tránh làm thay đổi kết quả đã hoàn tất. Các nút chức năng như “Xáo trộn”, “Giải bằng A Star” hoặc “Thoát” vẫn hoạt động để người chơi có thể bắt đầu ván mới hoặc thoát khỏi trò chơi.

Trạng thái kết thúc không chỉ đánh dấu việc hoàn thành một vòng chơi, mà còn là cơ hội để người chơi so sánh giải pháp thủ công với lời giải tự động của AI, từ đó hiểu rõ hơn về cách hoạt động của thuật toán A\*.



Hình 3.5 Hình ảnh giao diện chiến thắng trong trò chơi

## Một số kinh nghiệm trong trò chơi

Trong quá trình chơi và thử nghiệm trò chơi xếp hình 3x3, người chơi không chỉ phát triển kỹ năng quan sát và thao tác mà còn rút ra được nhiều chiến lược giải quyết bài toán một cách hiệu quả hơn. Các kinh nghiệm dưới đây đóng vai trò quan trọng trong việc nâng cao tốc độ giải và độ chính xác trong quá trình sắp xếp các mảnh ghép:

### Ưu tiên giải quyết theo từng hàng hoặc cột

Một trong những chiến lược cơ bản nhưng rất hiệu quả là chia nhỏ bài toán tổng thể thành các bài toán con – cụ thể là giải từng hàng hoặc từng cột một cách tuần tự. Ví dụ, người chơi nên bắt đầu bằng việc đưa ba ô đầu tiên (ô 1 đến ô 3) vào đúng vị trí của hàng đầu tiên, sau đó tiếp tục với các ô của hàng thứ hai (ô 4 đến ô 6), và cuối cùng xử lý ba ô còn lại cùng với ô trống. Cách tiếp cận phân tầng này giúp:

* Giảm độ phức tạp tạm thời của không gian trạng thái.
* Tránh làm nhiễu loạn từ các khu vực đã hoàn thiện.
* Hỗ trợ việc kiểm soát luồng di chuyển, đặc biệt trong giai đoạn cuối của trò chơi.

### Hạn chế làm rối phần đã hoàn thành

Việc di chuyển các ô ảnh một cách thiếu kiểm soát có thể làm xáo trộn lại những phần hình ảnh đã đúng vị trí. Người chơi nên đặt ra "vùng an toàn" – tức là những khu vực đã hoàn thiện – và hạn chế tối đa việc thao tác trong vùng đó. Điều này có có thể giúp:

* Tiết kiệm thời gian, tránh lặp lại các bước đã thực hiện.
* Giữ vững cấu trúc ổn định, hạn chế rơi vào vòng lặp không lối thoát.
* Tối ưu hóa chiến lược tổng thể, từ đó cải thiện hiệu suất giải đố.

### Sử dụng ô trống thông minh

Ô trống (Blank tile) là công cụ chính để thực hiện các thao tác di chuyển. Người chơi nên học cách điều phối ô trống sao cho luôn nằm ở vị trí thuận lợi nhất để thực hiện các bước di chuyển tiếp theo.

Một số chiến thuật thường được áp dụng như sau:

* Giữ ô trống gần khu vực đang thao tác, giúp giảm số bước di chuyển không cần thiết.
* Dùng ô trống để luân chuyển các mảnh xung quanh theo hướng thuận, tránh tạo các trạng thái "mắc kẹt".
* Tận dụng góc dưới bên phải (toạ độ(2,2)) làm điểm neo giúp dễ kiểm soát hướng quay của các mảnh ghép.

### Kiên nhẫn và đánh giá kỹ bước đi

Do bản chất của trò chơi là một bài toán tìm kiếm trong không gian trạng thái hữu hạn, mỗi bước đi đều ảnh hưởng đến cấu trúc tổng thể. Một bước đi sai có thể dẫn đến việc đảo lộn trật tự đã thiết lập, gây mất thời gian khôi phục. Do đó, người chơi cần:

* Dừng lại để quan sát trạng thái hiện tại, đặc biệt trước các bước rẽ nhánh.
* Dự đoán kết quả của thao tác kế tiếp, không chỉ nhìn vào trạng thái kế cận mà còn cần xét đến 2–3 bước tiếp theo.
* Kiên trì, vì đôi khi đường đi ngắn nhất không phải là đường trực tiếp mà là đường vòng chiến lược.

### Tận dụng thuật toán A Star

Đối với người chơi mới hoặc đang học về trí tuệ nhân tạo, chức năng giải tự động bằng thuật toán A\* trong trò chơi là một công cụ học tập hiệu quả. Người chơi có thể quan sát các bước giải của A\*, từ đó hiểu rõ cách thuật toán đánh giá trạng thái, chọn bước đi và tối ưu hành trình. Điều này giúp nâng cao tư duy logic và kỹ năng giải quyết bài toán trạng thái.

Khi người chơi chọn nút "Giải bằng A Star", người chơi có thể:

* Quan sát trực tiếp lộ trình tối ưu từ trạng thái hiện tại đến trạng thái đích.
* Hiểu rõ khái niệm hàm đánh giá với công thức f(n)=g(n)+h(n), trong đó g(n) là chi phí đã đi và h(n) là chi phí dự đoán còn lại.
* Học cách tư duy có chiến lược khi đối mặt với các vấn đề phức tạp – một kỹ năng cần thiết trong cả lập trình, giải thuật, và đời sống.

# ĐÁNH GIÁ VÀ HƯỚNG PHÁT TRIỂN

## Đánh giá

### Ưu điểm của trò chơi

Trò chơi xếp hình 3x3 sử dụng thuật toán A\* có một số ưu điểm nổi bật:

* Giao diện trực quan, dễ sử dụng: Người chơi có thể dễ dàng làm quen với giao diện nhờ bố cục rõ ràng, thao tác đơn giản với các nút chức năng cơ bản như “Xáo trộn”, “Giải bằng A Star”, “Âm thanh”,“Thoát”.
* Tăng cường tư duy logic và khả năng quan sát: Việc giải bài toán xếp hình đòi hỏi sự kiên nhẫn, khả năng phân tích và xây dựng chiến lược di chuyển, giúp người chơi rèn luyện kỹ năng tư duy.
* Ứng dụng trí tuệ nhân tạo vào thực tiễn: Việc tích hợp thuật toán tìm kiếm A\* giúp người chơi hiểu rõ hơn về cách hoạt động của AI trong việc tìm kiếm lời giải, từ đó kết nối giữa lý thuyết và thực hành.
* Khả năng mở rộng cao: Cấu trúc mã nguồn và thiết kế trò chơi cho phép dễ dàng phát triển thêm các tính năng mới như thay đổi hình nền, nâng cấp kích thước lưới hoặc chuyển sang nền tảng khác như web, mobile.

### Hạn chế của trò chơi

Bên cạnh những ưu điểm, trò chơi vẫn tồn tại một số điểm hạn chế cần khắc phục như sau:

* Chỉ hỗ trợ kích thước 3x3: Hiện tại, trò chơi chỉ dừng lại ở lưới 3x3 (8-puzzle), trong khi các phiên bản mở rộng như 4x4 hoặc 5x5 sẽ giúp tăng mức độ thử thách và hấp dẫn hơn.
* Giao diện còn đơn giản: Mặc dù chức năng khá đầy đủ, nhưng giao diện chưa được đầu tư về mặt thẩm mỹ như hiệu ứng chuyển động, thiết kế hình ảnh bắt mắt hoặc điều hướng thân thiện.
* Thiếu các tính năng bổ trợ: Trò chơi hiện chưa hỗ trợ tính năng lưu ván chơi cũ, chơi lại, đếm thời gian chơi hoặc tính điểm – những yếu tố quan trọng để tăng tính hấp dẫn và cạnh tranh.
* Thuật toán AI mới chỉ chạy theo yêu cầu: Người chơi phải tự bấm nút để giải bằng A\*, chưa có chức năng cho phép AI tự giải toàn bộ một cách trực tiếp hoặc hướng dẫn từng bước.

## Hướng phát triển

Trong tương lai, trò chơi có thể được mở rộng và cải tiến theo một số hướng như sau:

* Mở rộng kích thước lưới: Thêm tùy chọn lưới 4x4 (15-puzzle), 5x5… để tăng độ khó, giúp người chơi có thêm thử thách mới.
* Tùy chỉnh hình ảnh và giao diện: Cho phép người dùng chọn hình nền cá nhân (từ ảnh thư viện), sử dụng các chủ đề khác nhau cho các mảnh ghép, cải thiện trải nghiệm chơi.
* Tích hợp chế độ đua thời gian: Thêm đồng hồ bấm giờ, tính năng đua với thời gian để tăng tính thử thách và cạnh tranh giữa người chơi.
* Cho phép AI tự động giải hoàn toàn: Phát triển chế độ cho phép hệ thống tự giải toàn bộ lưới sau khi xáo trộn, có thể kèm theo hướng dẫn trực quan từng bước di chuyển.
* Thêm hệ thống điểm số và bảng xếp hạng: Ghi nhận thời gian hoàn thành, số bước di chuyển, từ đó tạo ra điểm số và so sánh giữa các người chơi.
* Triển khai trên nền tảng khác: Phát triển trò chơi thành phiên bản web hoặc mobile (Android/iOS) để tiếp cận nhiều người dùng hơn, sử dụng Unity như là công cụ chính trong phát triển đa nền tảng.

# KẾT LUẬN

Trong quá trình thực hiện đề tài “Xây dựng trò chơi xếp hình”, nhóm đã tiến hành đầy đủ các bước từ phân tích bài toán, thiết kế giao diện người dùng, lập trình logic trò chơi, đến việc tích hợp thuật toán tìm kiếm A\* nhằm giải quyết bài toán tìm đường tối ưu trong không gian trạng thái hữu hạn. Kết quả thu được cho thấy mô hình đã hoạt động ổn định, đáp ứng các yêu cầu đặt ra cả về mặt chức năng lẫn khả năng minh họa cho ứng dụng trí tuệ nhân tạo trong thực tiễn.

Cụ thể, sản phẩm hoàn chỉnh gồm các thành phần chính: giao diện trực quan, cơ chế xáo trộn hình ảnh đảm bảo tính khả giải, chức năng điều khiển bằng thao tác kéo thả các mảnh ghép, xác định trạng thái mục tiêu, cùng khả năng tự động giải bài toán sắp xếp bằng thuật toán A\*. Việc triển khai thuật toán A\* đã được thực hiện hiệu quả với hàm đánh giá f(n) = g(n) + h(n), trong đó g(n) là chi phí từ trạng thái ban đầu đến trạng thái hiện tại, còn h(n) là chi phí ước lượng từ trạng thái hiện tại đến trạng thái đích. Kết quả mô phỏng cho thấy thuật toán tìm được lời giải tối ưu với số bước di chuyển hợp lý.

Thông qua quá trình xây dựng và thử nghiệm trò chơi, nhóm đã rút ra nhiều bài học giá trị. Về mặt kỹ thuật, đề tài giúp củng cố kiến thức về cấu trúc dữ liệu, giải thuật tìm kiếm, thiết kế hệ thống tương tác và lập trình hướng đối tượng. Về mặt lý thuyết, đề tài giúp làm rõ cách thức hoạt động của các thuật toán heuristic và vai trò của chúng trong việc tối ưu hóa quá trình tìm kiếm lời giải trong các không gian trạng thái lớn.

Đáng chú ý, đề tài đã chứng minh được tính khả thi và giá trị ứng dụng của AI trong các hệ thống đơn giản như trò chơi giải đố. Sự kết hợp giữa yếu tố giải trí và yếu tố trí tuệ nhân tạo không chỉ tăng tính hấp dẫn của sản phẩm, mà còn mang lại tiềm năng lớn trong việc giáo dục tư duy thuật toán cho người học. Trò chơi không chỉ dừng lại ở mục đích giải trí, mà còn có thể được sử dụng như một công cụ trực quan hỗ trợ giảng dạy các môn học như tin học, trí tuệ nhân tạo cơ bản, hoặc lý thuyết tìm kiếm.

Trong tương lai, nhóm đề xuất một số hướng mở rộng như: tăng kích thước lưới trò chơi (4x4, 5x5), thêm các chế độ chơi nâng cao như đua thời gian, tích hợp hệ thống xếp hạng, và triển khai trên nền tảng web hoặc thiết bị di động để nâng cao khả năng tiếp cận của sản phẩm. Những phát triển này sẽ tiếp tục khai thác hiệu quả việc ứng dụng AI vào các mô hình trò chơi thông minh, góp phần làm phong phú hơn lĩnh vực giao thoa giữa giáo dục, giải trí và công nghệ.

# DANH MỤC TÀI LIỆU THAM KHẢO

[1] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 3rd ed. Pearson, 2010.

[2] Unity Technologies, *Unity - Real-time Development Platform*, 2024. [Online]. Available: <https://unity.com/>.

[3] “A\* search algorithm,” *Wikipedia*, [Online]. Available: [https://en.wikipedia.org/wiki/A\*\_search\_algorithm](https://en.wikipedia.org/wiki/A*_search_algorithm). [Accessed: 18-May-2025].

[4] Microsoft Docs, “C# Programming Guide,” Microsoft, 2024. [Online]. Available: https://learn.microsoft.com/en-us/dotnet/csharp/