|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**  **KHOA CÔNG NGHỆ THÔNG TIN**  **---------------------**    **ĐỒ ÁN**  **TỐT NGHIỆP ĐẠI HỌC**  ***Đề tài:***  **“Xây dựng trò chơi ứng dụng thiết bị nhận diện cử chỉ bàn tay Leap Motion Controller trên Unity Engine”**   |  |  | | --- | --- | | **Giảng viên hướng dẫn** | **: ThS. BÙI VĂN KIÊN** | | **Sinh viên thực hiện** | **: NGUYỄN TIẾN HÙNG** | | **Lớp** | **: D20CNPM05** | | **Mã sinh viên** | **: B20DCCN297** | | **Khóa** | **: 2020-2025** | | **Hệ** | **: ĐH CHÍNH QUY** |   *Hà Nội, 12/2024* |

# LỜI CẢM ƠN

Lời đầu tiên em xin gửi lời cảm ơn đến ThS. Bùi Văn Kiên, người đã tận tình hướng dẫn và đồng hành cùng em trong suốt quá trình thực hiện đồ án tốt nghiệp. Nhờ sự góp ý, chỉ dẫn nhiệt tình của thầy, em đã có thể triển khai và hoàn thiện tốt nhất các nội dung trong đề tài của mình.

Tiếp theo, em xin gửi lời cảm ơn chân thành tới ThS. Nguyễn Đức Hoàng, người đã định hướng đề tài, hỗ trợ thiết bị cần thiết và kết nối tới các anh chị có kinh nghiệm giúp em có được điều kiện thuận lợi nhất để thực hiện đồ án. Sự hỗ trợ của thầy đã giúp em vượt qua nhiều khó khăn trong quá trình nghiên cứu và thực hiện đề tài.

Ngoài ra, em cũng xin bày tỏ lòng biết ơn tới các thầy cô giáo trong Học viện Công nghệ Bưu chính Viễn thông, đặc biệt là các thầy cô khoa Công nghệ thông tin I, đã luôn tận tụy giảng dạy, truyền đạt kiến thức và kỹ năng cần thiết để em hoàn thành đồ án tốt nghiệp cũng như chuẩn bị hành trang cho công việc tương lai.

Cuối cùng, em xin gửi lời cảm ơn sâu sắc đến gia đình và bạn bè đã luôn động viên, ủng hộ và là nguồn động lực lớn lao cho em trong suốt thời gian thực hiện đồ án.

Trong quá trình thực hiện, dù đã cố gắng hết sức nhưng chắc chắn không tránh khỏi những thiếu sót. Em rất mong nhận được sự góp ý của thầy cô và các bạn để đề tài của em được hoàn thiện hơn.

Em xin chân thành cảm ơn!

Hà Nội, ngày 01 tháng 12 năm 2024

Nguyễn Tiến Hùng

# MỤC LỤC

[LỜI CẢM ƠN 2](#_Toc185204397)

[MỤC LỤC 3](#_Toc185204398)

[DANH MỤC BẢNG BIỂU 4](#_Toc185204399)

[DANH MỤC HÌNH ẢNH 6](#_Toc185204400)

[DANH MỤC CÁC KÝ HIỆU VÀ CHỮ VIẾT TẮT 12](#_Toc185204401)

[**1.** **TỔNG QUAN VỀ CÔNG NGHỆ NHÂN DIỆN CỬ CHỈ BÀN TAY VÀ LEAP MOTION CONTROLLER** 14](#_Toc185204402)

[**1.1.** **Công nghệ nhận diện cử chỉ bàn tay** 14](#_Toc185204403)

[**1.1.1.** **Bài toán chung** 14](#_Toc185204404)

[**1.1.2.** **Nguyên lý hoạt động** 14](#_Toc185204405)

[**1.1.3.** **Ứng dụng thực tế** 16](#_Toc185204406)

[**1.2.** **Leap Motion Controller** 18](#_Toc185204407)

[**1.2.1.** **Giới thiệu** 18](#_Toc185204408)

[**1.2.2.** **Cấu tạo và nguyên lý hoạt động** 20](#_Toc185204409)

[**1.2.3.** **Lịch sử phát triển** 22](#_Toc185204410)

[**1.2.4.** **Các ứng dụng** 24](#_Toc185204411)

[**2.** **GIỚI THIỆU UNITY ENGINE VÀ ULTRALEAP PLUGIN** 27](#_Toc185204412)

[**2.1.** **Giới thiệu về Unity Engine** 27](#_Toc185204413)

[**2.1.1.** **Tổng quan** 27](#_Toc185204414)

[**2.1.2.** **Mục đích hình thành và phát triển** 28](#_Toc185204415)

[**2.1.3.** **Ưu và nhược điểm của Unity** 29](#_Toc185204416)

[**2.1.4.** **Một số thành phần chính trong Unity Editor** 30](#_Toc185204417)

[**2.1.5.** **Một số khái niệm trong Unity Editor** 33](#_Toc185204418)

[**2.2.** Giới thiệu về Ultraleap plugin 34](#_Toc185204419)

[2.2.1. Tổng quan về Ultraleap plugin 34](#_Toc185204420)

[2.2.2. Các thành phần chính trong Ultraleap 34](#_Toc185204421)

[**2.3.** Tích hợp Ultraleap vào Unity 34](#_Toc185204422)

[2.3.1. Chuẩn bị môi trường làm việc 34](#_Toc185204423)

[2.3.2. Cài đặt Ultraleap plugin 34](#_Toc185204424)

[2.3.3. Các ứng dụng của Ultraleap trong phát triển game 34](#_Toc185204425)

[**2.4.** Các vấn đề thường gặp và cách giải quyết 34](#_Toc185204426)

[**3.** **PHÂN TÍCH VÀ THIÊT KẾ GAME CHÉM HOA QUẢ** 34](#_Toc185204427)

[**3.1.** Biểu đồ phân rã chức năng 34](#_Toc185204428)

[**3.2.** Biểu đồ usecase 34](#_Toc185204429)

[**3.3.** Kịch bản usecase 34](#_Toc185204430)

[**3.4.** Biểu đồ lớp 34](#_Toc185204431)

[**3.5.** Biểu đồ tuần tự 35](#_Toc185204432)

[**4.** **XÂY DỰNG GAME CHÉM HOA QUẢ CÓ TÍCH HỢP LEAP MOTION VỚI UNITY ENGINE** 35](#_Toc185204433)

[**4.1.** Xây dựng hệ thống tạo hoa quả 35](#_Toc185204434)

[**4.2.** Tích hợp Ultraleap 35](#_Toc185204435)

[**4.3.** Xử lý logic game 35](#_Toc185204436)

[**4.4.** Cài đặt giao diện cơ bản 35](#_Toc185204437)

[**4.5.** 4 hoá và kiểm thử 35](#_Toc185204438)

[**5.** **TÀI LIỆU THAM KHẢO** 35](#_Toc185204439)

# DANH MỤC BẢNG BIỂU

[Bảng 2.4.1: Bảng chỉ số áp dụng cho tất cả unit trong game 48](#_Toc182321846)

[Bảng 2.4.2: Bảng chỉ số áp dụng riêng cho Hero 48](#_Toc182321847)

[Bảng 2.4.3: Bảng chỉ số áp dụng riêng cho Troop 48](#_Toc182321848)

[Bảng 2.4.4: Bảng chỉ số áp dụng riêng cho Enemy 48](#_Toc182321849)

[Bảng 2.4.5: Bảng chỉ số áp dụng riêng cho Boss và Miniboss 48](#_Toc182321850)

[Bảng 2.4.6: Bảng tính độ khó chó 20 Level 62](#_Toc182321851)

[Bảng 2.4.7: Bảng kịch bản tutorial 1 66](#_Toc182321852)

[Bảng 2.4.8: Bảng kịch bản tutorial 2 67](#_Toc182321853)

[Bảng 2.4.9: Bảng kịch bản tutorial 3 67](#_Toc182321854)

[Bảng 3.1.1: Kịch bản chơi game 93](#_Toc182321855)

[Bảng 3.1.2: Kịch bản nâng cấp quân 93](#_Toc182321856)

[Bảng 3.1.3: Kịch bản nhận thưởng thành tựu 93](#_Toc182321857)

[Bảng 3.1.4: Kịch bản cài đặt 94](#_Toc182321858)

[Bảng 3.1.5: Kịch bản triệu hồi quân 94](#_Toc182321859)

# DANH MỤC HÌNH ẢNH

[Hình 1.1.1: Các thiết bị chơi game 17](#_Toc182822868)

[Hình 1.1.2. Biểu đồ thị phần các loại game 18](#_Toc182822869)

[Hình 1.2.1: Game Dead Space 19](#_Toc182822870)

[Hình 1.2.2: Game Devil May Cry 20](#_Toc182822871)

[Hình 1.2.3: Game Street Fighter 21](#_Toc182822872)

[Hình 1.2.4: Game Counter-Strike 1.6 22](#_Toc182822873)

[Hình 1.2.5: Game Mario 23](#_Toc182822874)

[Hình 1.2.6: Game Tetris 24](#_Toc182822875)

[Hình 1.2.7: Game Runescape 25](#_Toc182822876)

[Hình 1.2.8: Game Rogue 26](#_Toc182822877)

[Hình 1.2.9: Game Marvel Snap 27](#_Toc182822878)

[Hình 1.2.10: Game League of Legend 28](#_Toc182822879)

[Hình 1.2.11: Game Kingdom Rush 29](#_Toc182822880)

[Hình 1.3.1. Logo Unity 29](#_Toc182822881)

[Hình 1.3.2. Cửa sổ Scene 33](#_Toc182822882)

[Hình 1.3.3. Cửa sổ Hierarchy trong Unity 34](#_Toc182822883)

[Hình 1.3.4. Cửa sổ Game trong Unity 34](#_Toc182822884)

[Hình 1.3.5. Cửa sổ Project trong Unity 35](#_Toc182822885)

[Hình 1.3.6. Cửa sổ Inspector trong Unity 35](#_Toc182822886)

[Hình 1.4.1: Game Blackmyth Wukong 37](#_Toc182822887)

[Hình 1.4.2: Thị phần doanh thu game toàn cầu năm 2023 37](#_Toc182822888)

[Hình 1.4.3: Tỉ lệ thiết bị người Việt dùng để chơi game 38](#_Toc182822889)

[Hình 2.1.1: Game Bad North 40](#_Toc182822890)

[Hình 2.1.2: Workflow cơ bản của team 41](#_Toc182822891)

[Hình 2.4.1: Hexagonal Grid 43](#_Toc182822892)

[Hình 2.4.2: Bản đồ xứ Sylvan 47](#_Toc182822893)

[Hình 2.4.3: Bản đồ Verdant Isle 47](#_Toc182822894)

[Hình 2.4.4: Bản đồ Spiritguider Isle 48](#_Toc182822895)

[Hình 2.4.5: Bản đồ Vineweaver Land 48](#_Toc182822896)

[Hình 2.4.6 Bản đồ Stalkborne Farland 49](#_Toc182822897)

[Hình 2.4.7: Bản đồ Hawthorne Island 49](#_Toc182822898)

[Hình 2.4.8: Bản đồ Citrusguard Island 50](#_Toc182822899)

[Hình 2.4.9: Bản đồ Leafborne Vale 50](#_Toc182822900)

[Hình 2.4.10: Bản đồ Rootkin Dryland 51](#_Toc182822901)

[Hình 2.4.11: Bản đồ Chestnut Isle 51](#_Toc182822902)

[Hình 2.4.12: Layout Main Menu 52](#_Toc182822903)

[Hình 2.4.13: Layout trong game 52](#_Toc182822904)

[Hình 2.4.14: Layout Xem danh sách Troop 53](#_Toc182822905)

[Hình 2.4.15: Layout Chi tiết Troop 53](#_Toc182822906)

[Hình 2.4.16: Layout Nâng cấp Troop trong game 54](#_Toc182822907)

[Hình 2.4.17: Layout codex 54](#_Toc182822908)

[Hình 2.4.18: Layout Achievement 55](#_Toc182822909)

[Hình 2.4.19: Layout Cài đặt 55](#_Toc182822910)

[Hình 2.4.20: Màn hình Main Menu 56](#_Toc182822911)

[Hình 2.4.21: Màn hình trong game 56](#_Toc182822912)

[Hình 2.4.22: Công thức hàm Sigmoid 57](#_Toc182822913)

[Hình 2.4.23: Biểu đồ Độ khó và Số Fragment khởi đầu 59](#_Toc182822914)

[Hình 2.4.24: Thiết kế Level 1 60](#_Toc182822915)

[Hình 2.4.25: Thiết kế Level 2 60](#_Toc182822916)

[Hình 2.4.26: Thiết kế Level 3 61](#_Toc182822917)

[Hình 2.4.27: Thiết kế Level 4 61](#_Toc182822918)

[Hình 2.4.28: Thiết kế Level 5 62](#_Toc182822919)

[Hình 2.4.29: Danh sách các Hero hiện có 64](#_Toc182822920)

[Hình 2.4.30: Bảng lên cấp của Hero 65](#_Toc182822921)

[Hình 2.4.31: Bảng Boon của Hero 66](#_Toc182822922)

[Hình 2.4.32: Danh sách các Troop 67](#_Toc182822923)

[Hình 2.4.33: Bảng nâng cấp Troop 67](#_Toc182822924)

[Hình 2.4.34: Bảng Boon của Troop 68](#_Toc182822925)

[Hình 2.4.35: Danh sách các Enemy 69](#_Toc182822926)

[Hình 2.4.36: Bảng nâng cấp của Enemy 69](#_Toc182822927)

[Hình 2.4.37: Danh sách Miniboss 70](#_Toc182822928)

[Hình 2.4.38: Danh sách Boss 70](#_Toc182822929)

[Hình 2.4.39: Bảng phân bổ độ khó cho Phase và Wave 71](#_Toc182822930)

[Hình 2.4.40: Các tham số của Active Skill 72](#_Toc182822931)

[Hình 2.4.41: Bảng các Active Skill của Hero 72](#_Toc182822932)

[Hình 2.4.42: Bảng các Active Skill của Miniboss 72](#_Toc182822933)

[Hình 2.4.43: Bảng các Active Skill của Boss 72](#_Toc182822934)

[Hình 2.4.44: Bảng các Passive Skill 73](#_Toc182822935)

[Hình 2.4.45: Bảng các Spell 73](#_Toc182822936)

[Hình 2.4.46: Bảng các hiệu ứng đặc biệt 73](#_Toc182822937)

[Hình 2.4.47: Danh sách các Achievement 74](#_Toc182822938)

[Hình 2.4.48: Moodboard quân Spriteling 75](#_Toc182822939)

[Hình 2.4.49: Moodboard quân Sapling Sentry 75](#_Toc182822940)

[Hình 2.4.50: Moodboard quân Vine Beast 75](#_Toc182822941)

[Hình 2.4.51: Moodboard quân Brocolance 75](#_Toc182822942)

[Hình 2.4.52: Moodboard của Jorath the Thorned 76](#_Toc182822943)

[Hình 2.4.53: Moodboard của Asparagon 76](#_Toc182822944)

[Hình 2.4.54: Moodboard của Carrotus 76](#_Toc182822945)

[Hình 2.4.55: Moodboard của Zulma the Witch 76](#_Toc182822946)

[Hình 2.4.56: Moodboard của Captain Ora 77](#_Toc182822947)

[Hình 2.4.57: Moodboard của Tendra Shield Bug 77](#_Toc182822948)

[Hình 2.4.58: Moodboard của Tendra Eagle 77](#_Toc182822949)

[Hình 2.4.59: Moodboard của Tendra Gekko 77](#_Toc182822950)

[Hình 2.4.60: Moodboard của Tendra Apis 78](#_Toc182822951)

[Hình 2.4.61: Moodboard của Tendra Mandibite 78](#_Toc182822952)

[Hình 2.4.62: Moodboard của Tendra Brood Mother 78](#_Toc182822953)

[Hình 2.4.63: Moodboard của They who Swallow 78](#_Toc182822954)

[Hình 2.4.64: Moodboard của They who Launch 79](#_Toc182822955)

[Hình 2.4.65: Moodboard của They who Chase 79](#_Toc182822956)

[Hình 2.4.66: Moodboard của They who Decay 79](#_Toc182822957)

[Hình 2.4.67: Moodboard của They who Endure 79](#_Toc182822958)

[Hình 2.4.68: Moodboard của They who Command 80](#_Toc182822959)

[Hình 2.4.69: Moodboard của Shell Bearer 80](#_Toc182822960)

[Hình 2.4.70: Moodboard của Reef Warrior 80](#_Toc182822961)

[Hình 2.4.71: Moodboard của Tide Breaker 80](#_Toc182822962)

[Hình 2.4.72: Moodboard của Sea Drifter 81](#_Toc182822963)

[Hình 2.4.73: Moodboard của Water Gusher 81](#_Toc182822964)

[Hình 2.4.74: Moodboard của Abyss Cultist 81](#_Toc182822965)

[Hình 2.4.75: Moodboard của Brungus 81](#_Toc182822966)

[Hình 2.4.76: Moodboard của Shroom Blade 82](#_Toc182822967)

[Hình 2.4.77: Moodboard của Spore Slapper 82](#_Toc182822968)

[Hình 2.4.78: Moodboard của Shroom Thrower 82](#_Toc182822969)

[Hình 2.4.79: Moodboard của Spore Shooter 82](#_Toc182822970)

[Hình 2.4.80: Moodboard của Ancient Mycelia 83](#_Toc182822971)

[Hình 2.4.81: Moodboard của The First Tree 83](#_Toc182822972)

[Hình 2.4.82: Moodboard của The Verdant Kin 83](#_Toc182822973)

[Hình 2.4.83: Moodboard của Tendra’s Camp 83](#_Toc182822974)

[Hình 2.4.84: Moodboard của Seasong’s Camp 84](#_Toc182822975)

[Hình 2.4.85: Moodboard của Starboom’s Camp 84](#_Toc182822976)

[Hình 2.4.86: Moodboard của Mycelia’s Camp 84](#_Toc182822977)

[Hình 3.1.1: Biểu đồ phân rã chức năng 85](#_Toc182822978)

[Hình 3.1.2: Usecase tổng quan 86](#_Toc182822979)

[Hình 3.1.3: Usecase chơi game 87](#_Toc182822980)

[Hình 3.1.4: Usecase Cài đặt 87](#_Toc182822981)

[Hình 3.1.5: Usecase Thành tựu 88](#_Toc182822982)

[Hình 3.1.6: Usecase Nâng cấp quân 88](#_Toc182822983)

[Hình 3.1.7: Biểu đồ lớp Hero 90](#_Toc182822984)

[Hình 3.1.8: Biểu đồ lớp Troop 91](#_Toc182822985)

[Hình 3.1.9: Biểu đồ lớp Enemy 91](#_Toc182822986)

[Hình 3.1.10: Biểu đồ lớp Spell 92](#_Toc182822987)

[Hình 3.1.11: Biểu đồ tuần tự chơi game 92](#_Toc182822988)

[Hình 3.1.12: Biểu đồ tuần tự nâng cấp quân 93](#_Toc182822989)

[Hình 3.1.13: Biểu đồ tuần tự triệu hồi quân 93](#_Toc182822990)

[Hình 3.1.14: Biểu đồ tuần tự nhận thưởng thành tựu 94](#_Toc182822991)

[Hình 3.2.1: Model nhà chính 94](#_Toc182822992)

[Hình 3.2.2: Bộ model các Tile lục giác 95](#_Toc182822993)

[Hình 3.2.3: Bộ model các cây cỏ 95](#_Toc182822994)

[Hình 3.2.4: Bộ model các khối đá 96](#_Toc182822995)

[Hình 3.2.5: Chế độ Vertex Snapping 96](#_Toc182822996)

[Hình 3.2.6: Level hoàn thiện 97](#_Toc182822997)

[Hình 3.2.7: Hierarchy của một Level 97](#_Toc182822998)

[Hình 3.2.8: Unity AI Navigation package 98](#_Toc182822999)

[Hình 3.2.9: Nhân vật với Nav Mesh Agent 99](#_Toc182823000)

[Hình 3.2.10: NavMeshSurface 100](#_Toc182823001)

[Hình 3.2.11: NavMesh 101](#_Toc182823002)

[Hình 3.2.12: Các tham số của Script NavCharBase 101](#_Toc182823003)

[Hình 3.2.13: Hàm Start và Update của NavCharBase 102](#_Toc182823004)

[Hình 3.2.14: Các tham số của NavCharRandom 103](#_Toc182823005)

[Hình 3.2.15: Hàm Update của NavCharRandom 103](#_Toc182823006)

[Hình 3.3.1: Hierarchy Camera trong Unity 104](#_Toc182823007)

[Hình 3.3.2: Các tham số trong script CameraController 104](#_Toc182823008)

[Hình 3.3.3: Hàm Start của script CameraController 105](#_Toc182823009)

[Hình 3.3.4: Hàm Update và LateUpdate của script CameraController 105](#_Toc182823010)

[Hình 3.3.5: Hàm HandleMouseInput xử lý Zoom Camera 105](#_Toc182823011)

[Hình 3.3.6: Hàm HandleMouseInput xử lý Move Camera 106](#_Toc182823012)

[Hình 3.3.7: Hàm HandleMouseInput xử lý Rotate Camera 107](#_Toc182823013)

[Hình 3.3.8: Màn hình game và Hierarchy cơ chế Spawn Enemy 108](#_Toc182823014)

[Hình 3.3.9: Scriptable Object của Enemy 108](#_Toc182823015)

[Hình 3.3.10: Các Enemy SO khác nhau 109](#_Toc182823016)

[Hình 3.3.11: Các gameobject text trong script WaveManager 109](#_Toc182823017)

[Hình 3.3.12: Các giá trị dùng để tính toán độ khó trong script WaveManager 110](#_Toc182823018)

[Hình 3.3.13: Hàm Start của script WaveManager 110](#_Toc182823019)

[Hình 3.3.14: Hàm Update và SetUI của script WaveManager 111](#_Toc182823020)

[Hình 3.3.15: Hàm CalculatePhase của script WaveManager 111](#_Toc182823021)

[Hình 3.3.16: Hàm CalculateDifficutly của script WaveManager 112](#_Toc182823022)

[Hình 3.3.17: Các tham số của script SpawnEnemy 113](#_Toc182823023)

[Hình 3.3.18: Hàm start của script SpawnEnemy 113](#_Toc182823024)

[Hình 3.3.19: Hàm InitializeSpawnableEnemies() trong script SpawnEnemy 113](#_Toc182823025)

[Hình 3.3.20: Hàm Update của script SpawnEnemy 114](#_Toc182823026)

[Hình 3.3.21: Hàm LoadSpawnableEnemy của script SpawnEnemy 114](#_Toc182823027)

[Hình 3.3.22: Hàm SpawnSingleEnemie của script SpawnEnemy 115](#_Toc182823028)

# DANH MỤC CÁC KÝ HIỆU VÀ CHỮ VIẾT TẮT

|  |  |
| --- | --- |
| Fps: Frame per second | Số khung hình trên giây |
| AI: Artificial Intelligence | Trí tuệ nhân tạo |
| Co-op: Co-operative | Phối hợp |
| FPS: First Person Shooter | Bắn súng góc nhìn thứ nhất |
| TPS: Third Person Shooter | Bắn súng góc nhìn thứ ba |
| MMO: Massive Multiplayer Online | Game nhiều người chơi online |
| RPG: Role Play Game | Game nhập vai |
| TD: Tower Defense | Game thủ thành |
| MOBA: Multiplayer Online Battle Arena | Game đầu trường online đa người chơi |
| PC: Personal Computer | Máy tính cá nhân |
| GD: Game Designer | Người thiết kế game |
| GDD: Game Design Document | Tài liệu thiết kế game |

**Lời mở đầu**

Trong bối cảnh xã hội tiến bộ ngày nay khi công nghệ được phát triển không ngừng, mỗi chúng ta hẳn không còn xa lạ gì với khái niệm công nghệ thực tế ảo VR-AR. Không chỉ mang lại những trải nghiệm mới mẻ mà nó còn mở ra nhiều tiềm năng ứng dụng đa dạng trong các lĩnh vực như giáo dục, y tế, giải trí, và đặc biệt là trò chơi điện tử. Leap Motion là thiết bị cảm biến chuyển động tiên tiến, cho phép người dùng tương tác với máy tính thông qua cử chỉ tay, mang đến sự kết hợp hài hòa giữa công nghệ hiện đại và tính trực quan trong thiết kế tương tác.

Đồ án tập trung khai thác tiềm năng của công nghệ này trong lĩnh vực trò chơi, với mục tiêu phát triển game Chém hoa quả (Fruit Ninja) – 1tựa game thành công và nổi tiếng toàn cầu, hẳn ai cũng đã từng chơi hoặc xem và hình dung ra được lối chơi của trò chơi này trên các thiết bị cảm ứng nhưng giờ đây nó sẽ được trải nghiệm theo 1 cách hoàn toàn khác.

Nội dung đồ án gồm các phần như sau

1. Chương 1: Tổng quan về công nghệ nhân diện cử chỉ bàn tay và Leap Motion Controller
2. Chương 2: Giới thiệu Unity Engine và Ultraleap plugin
3. Chương 3: Phân tích và thiết kế game Chém hoa quả

Chương 4: Xây dựng game Chém hoa quả sử dụng Leap Motion Controller

1. **TỔNG QUAN VỀ CÔNG NGHỆ NHÂN DIỆN CỬ CHỈ BÀN TAY VÀ LEAP MOTION CONTROLLER**
   1. **Công nghệ nhận diện cử chỉ bàn tay**
      1. **Bài toán chung**

Bài toán nhận diện cử chỉ tay đã được quan tâm và nghiên cứu từ những năm đầu thế kỷ trước. Mục tiêu là nhận diện được hình dáng (cử chỉ tay tĩnh) hoặc hành động (cử chỉ tay động) của cử chỉ tay, bao gồm ngón tay, bàn tay, cánh tay để từ đó đưa ra các thông tin hữu ích trong tương tác người – máy.

Trước đây bài toán nhận dạng cử chỉ tay thường được tiếp cận theo hướng áp dụng thị giác máy tính. Theo đó các cử chỉ ngón tay, bàn tay sẽ được chụp lại và sử dụng các mô hình học máy để huấn luyện và nhận dạng. Cách tiếp cận này đạt được độ chính xác cao trong nhận dạng các cử chỉ tay tĩnh nhờ khả năng xử lý tốt của các thuật toán máy học. Tuy nhiên, nó có những hạn chế nhất định, đặc biệt khi áp dụng vào nhận diện các cử chỉ tay động. Phạm vi nhận dạng bị giới hạn trong vùng nhìn của camera, và hệ thống thường đòi hỏi tài nguyên tính toán lớn, khiến nó không phù hợp với các ứng dụng yêu cầu độ phản hồi thời gian thực cao hoặc hoạt động trong môi trường đa dạng.

Với sự phát triển mạnh mẽ của công nghệ vi điện tử, đặc biệt trong thập kỷ gần đây, các cảm biến hiện đại đã mở ra những hướng tiếp cận mới cho bài toán nhận diện cử chỉ tay. Các thiết bị cảm biến đã vượt qua những giới hạn của thị giác máy tính truyền thống bằng cách sử dụng dữ liệu không gian ba chiều (3D) để phân tích chính xác các cử động của bàn tay và ngón tay. Công nghệ này không chỉ tăng cường độ chính xác trong nhận diện mà còn mở rộng phạm vi ứng dụng, cho phép người dùng thực hiện các thao tác ở nhiều vị trí và điều kiện môi trường khác nhau mà không cần camera trực diện. Kết hợp với các thuật toán tiên tiến như học sâu (Deep Learning), hệ thống hiện nay có thể không chỉ nhận diện cử chỉ tay tĩnh mà còn phân tích được các chuỗi hành động phức tạp theo thời gian.

* + 1. **Nguyên lý hoạt động**

Nguyên lý hoạt động của các thiết bị nhân diện cử chỉ bàn tay dựa trên việc thu thập dữ liệu từ bàn tay (vị trí, hình dạng, và chuyển động) sau đó phân tích để nhận diện các cử chỉ hoặc hành vi mong muốn.

Đối với các thiết bị dựa trên thị giác máy tính (Computer Vision-Based Systems), nó hoạt động bằng cách sử dụng một hoặc nhiều camera để ghi nhận hình ảnh hoặc video của bàn tay trong thời gian thực. Quá trình xử lý diễn ra qua các bước:

1. Ghi nhận hình ảnh: Camera thu thập hình ảnh của bàn tay. Một số thiết bị cao cấp sử dụng camera chiều sâu (depth camera), cho phép phân biệt giữa các đối tượng trong không gian ba chiều và cải thiện độ chính xác.
2. Xử lý hình ảnh: Các thuật toán xử lý ảnh sẽ tách biệt bàn tay ra khỏi nền bằng các kỹ thuật như phân đoạn hình ảnh (image segmentation) hoặc phát hiện biên (edge detection). Sau đó, bàn tay được trích xuất dưới dạng hình dạng 2D hoặc mô hình 3D.
3. Nhận diện cử chỉ: Dựa trên hình dạng hoặc chuyển động của bàn tay, các hệ thống sử dụng các mô hình học máy hoặc học sâu để nhận diện các cử chỉ tĩnh (như giơ tay, chỉ ngón) hoặc động (như vẫy tay, xoay cổ tay).

Hệ thống này phù hợp cho nhiều ứng dụng, nhưng hiệu quả có thể bị ảnh hưởng bởi môi trường ánh sáng hoặc sự che khuất của bàn tay

A collage of hands with green lines

Description automatically generated

Đối với nhóm thiết bị sử dụng cảm biến chuyên dụng (Sensor-Based Systems):

1. Thu thập dữ liệu từ cảm biến:

* Cảm biến hồng ngoại (Infrared Sensors): Thiết bị phát ra ánh sáng hồng ngoại và ghi nhận tín hiệu phản xạ từ bề mặt bàn tay để xác định vị trí, hình dạng, và độ sâu trong không gian 3D.
* Cảm biến siêu âm (Ultrasonic Sensors): Đo thời gian sóng âm phản xạ để xác định khoảng cách và hướng di chuyển..
* Cảm biến điện tử gắn trên tay (Wearable Sensors): Ghi nhận chuyển động của các khớp và độ nghiêng từ gia tốc kế, con quay hồi chuyển hoặc cảm biến lực.

1. Tái tạo mô hình bàn tay: Sử dụng dữ liệu để xây dựng mô hình xương và khớp bàn tay trong không gian 3D (skeleton model).

Nhóm thiết bị này có ưu điểm là hoạt động ổn định hơn trong nhiều môi trường và cho phép nhận diện cử chỉ tay động một cách chính xác hơn.

A diagram of a computer and a phone

Description automatically generated

* + 1. **Ứng dụng thực tế**

Công nghệ nhận diện cử chỉ bàn tay đang dần trở thành một phần không thể thiếu trong nhiều lĩnh vực khác nhau nhờ tính tương tác tự nhiên và không cần chạm.

Nó được sử dụng rộng rãi trong các thiết bị VR/AR để tăng cường trải nghiệm tương tác, giúp người dùng tương tác một cách tự nhiên với thế giới ảo mà không cần sử dụng tay cầm hay các công cụ ngoại vi khác. Một số sản phẩm VR/AR nổi bật có sử dụng công nghệ này như: Oculus Quest của Meta, Apple Vision Pro hay HoloLens của Microsoft, …

Trong y tế, việc sử dụng cử chỉ tay giúp các bác sĩ và nhân viên y tế tương tác với hệ thống mà không cần tiếp xúc vật lý, đặc biệt trong các phòng mổ, nơi yêu cầu vệ sinh cao. Các công ty như GestSure và Surgical Theater đang tiên phong trong việc ứng dụng công nghệ này.

* *GestSure*: cho phép bác sĩ điều khiển hình ảnh y tế trong khi phẫu thuật mà không cần chạm vào màn hình, giúp giảm thiểu nguy cơ nhiễm trùng và nâng cao hiệu quả công việc.
* *Surgical Theater*: sử dụng mô phỏng 3D để hỗ trợ bác sĩ trong việc lên kế hoạch phẫu thuật, công ty này tích hợp công nghệ nhận diện cử chỉ tay để người dùng có thể tương tác với các mô hình y tế mà không cần thiết bị ngoại vi.

A person in a surgical gown and gloves working on a computer

Description automatically generated

Trong ngành công nghiệp game, sự phát triển của các hệ thống tương tác không chạm đã tạo ra những trải nghiệm chơi game mới mẻ, trong đó người chơi có thể trực tiếp tham gia vào các hành động trong game thông qua cử chỉ tay.

* *Fruit Ninja Kinect (Xbox – Microsoft)*: cho phép người chơi sử dụng cử chỉ tay để chém trái cây trong không gian 3D mà không cần bất kỳ tay cầm nào, sử dụng công nghệ nhận diện cử chỉ tay của Kinect.
* *Leap Motion (Ultraleap)*: cho phép người dùng điều khiển các trò chơi VR hoặc các ứng dụng 3D bằng cử chỉ tay. Công nghệ này được tích hợp vào các game như *The Unspoken* và *Blocks*, mang lại trải nghiệm game trải nghiệm game tương tác hoàn toàn mới.

Công nghệ nhận diện cử chỉ tay cũng đã được ứng dụng trong giáo dục, đặc biệt là trong các lớp học ảo và mô phỏng, nơi học sinh và giáo viên có thể tương tác với các mô hình 3D mà không cần sử dụng thiết bị ngoại vi.

* *ClassVR:* nền tảng giáo dục này cung cấp những trải nghiệm học tập thực tế ảo, nơi học sinh có thể tương tác với các bài học và mô phỏng 3D chỉ qua cử chỉ tay, giúp tăng tính trực quan và sinh động cho quá trình học.
* *zSpace:* sử dụng công nghệ nhận diện cử chỉ tay để học sinh có thể tương tác với các mô hình 3D trong khi học các môn khoa học, kỹ thuật và nghệ thuật. Việc sử dụng các mô hình 3D này giúp học sinh dễ dàng tiếp cận các khái niệm phức tạp.

Trong lĩnh vực sản xuất và công nghiệp, các công ty lớn đã bắt đầu ứng dụng cử chỉ tay để điều khiển robot và dây chuyền sản xuất mà không cần tiếp xúc vật lý, giúp tăng cường hiệu quả công việc và giảm thiểu rủi ro.

Công nghệ nhận diện cử chỉ tay cũng đã được ứng dụng trong các thiết bị nhà thông minh, nơi người dùng có thể điều khiển các thiết bị trong nhà mà không cần sử dụng remote hay công tắc.

Với tiềm năng phát triển lớn, công nghệ này đang hướng đến việc kết hợp với trí tuệ nhân tạo (AI) để tạo ra các hệ thống tương tác thông minh hơn, có khả năng học hỏi và thích nghi với hành vi của từng cá nhân. Trong tương lai, các thiết bị nhận diện cử chỉ tay có thể được tích hợp sâu hơn vào các thiết bị di động, wearable, hoặc các thiết bị IoT, mở ra khả năng tương tác linh hoạt hơn và thúc đẩy sự phát triển của môi trường số hóa toàn diện. Những tiến bộ này hứa hẹn sẽ đưa công nghệ nhận diện cử chỉ bàn tay từ một công cụ hỗ trợ trở thành một thành phần cốt lõi trong tương tác giữa con người và máy móc.

* 1. **Leap Motion Controller**
     1. **Giới thiệu**

Leap Motion Controller là một thiết bị ngoại vi được thiết kế để theo dõi và nhận diện các chuyển động tay và ngón tay của người dùng trong không gian ba chiều. Thiết bị này được phát triển bởi Leap Motion Inc., một công ty công nghệ có trụ sở tại San Francisco, Mỹ. Với kích thước nhỏ gọn và khả năng theo dõi chính xác, Leap Motion Controller đã mở ra một hướng đi mới trong cách con người tương tác với máy tính và các hệ thống kỹ thuật số mà không cần sử dụng đến chuột, bàn phím hay các thiết bị điều khiển truyền thống.

A device with a black screen

Description automatically generated with medium confidence

Thiết bị nhỏ gọn và nhẹ, mang thiết kế hiện đại với vẻ ngoài cao cấp. Khung viền được chế tạo từ nhôm nguyên khối, các góc bo tròn tinh tế, và mặt trên là một tấm kính tối màu trong suốt, giúp thiết bị theo dõi chính xác các chuyển động tay trong không gian. Phần kính này bảo vệ hai cảm biến và đèn LED hồng ngoại, hỗ trợ thiết bị hoạt động bền bỉ và hiệu quả. Mặt đế được trang bị lớp cao su chống trượt với logo Leap Motion khắc chìm, đảm bảo sự ổn định khi sử dụng. Thiết bị được kết nối với máy tính qua cổng mini USB, đi kèm đèn LED hiển thị trạng thái hoạt động.

A close-up of a device

Description automatically generated

* + 1. **Cấu tạo và nguyên lý hoạt động**

Leap Motion Controller có thiết kế nhỏ gọn với kích thước chỉ 80 x 30 x 11.3 mm và trọng lượng rất nhẹ chỉ 32g.

A blue rectangular object with black text

Description automatically generated

Thiết bị bao gồm các thành phần chính sau:

* Cảm biến hồng ngoại (Infrared Sensors): Có hai cảm biến hồng ngoại nằm bên trong thiết bị, chịu trách nhiệm thu thập hình ảnh 3D của bàn tay và các vật thể nằm trong phạm vi theo dõi.
* Đèn LED hồng ngoại (Infrared LEDs): Ba đèn LED hồng ngoại được sử dụng để chiếu sáng vùng không gian mà thiết bị theo dõi. Ánh sáng hồng ngoại không gây ảnh hưởng đến mắt người nhưng lại giúp cảm biến ghi lại hình ảnh rõ ràng, ngay cả trong điều kiện ánh sáng yếu.
* Vi xử lý (Processor): Thiết bị tích hợp một vi xử lý để xử lý sơ bộ dữ liệu hình ảnh trước khi gửi tới máy tính qua cổng USB.
* Cổng kết nối USB: Kết nối giữa Leap Motion Controller và máy tính thông qua cổng USB, vừa cung cấp nguồn điện cho thiết bị, vừa truyền dữ liệu.

A rectangular device with a diagram

Description automatically generated with medium confidence

Nguyên lý hoạt động của Leap Motion dựa trên việc phát và thu ánh sáng hồng ngoại để xây dựng bản đồ 3D của các vật thể trong vùng quan sát:

* Các đèn LED phát ánh sáng hồng ngoại chiếu vào bàn tay hoặc các vật thể trong phạm vi hoạt động (tối đa khoảng 60 cm).
* Cảm biến hồng ngoại ghi nhận ánh sáng phản xạ lại từ bàn tay và các vật thể này.
* Dữ liệu từ cảm biến được chuyển đổi thành hình ảnh 3D và gửi tới máy tính.
* Phần mềm Leap Motion SDK xử lý dữ liệu để xác định vị trí, hình dạng và chuyển động của tay, từ đó nhận diện các cử chỉ như kéo, thả, vuốt, hoặc phóng to/thu nhỏ.

Nhờ vào thiết kế tinh vi và các thuật toán mạnh mẽ, Leap Motion Controller đạt độ chính xác cao đến từng milimet, với khả năng nhận diện chuyển động của từng ngón tay ngay cả khi chúng di chuyển nhanh.

* + 1. **Lịch sử phát triển**

Leap Motion Controller được giới thiệu lần đầu tiên vào năm 2012 và chính thức ra mắt vào năm 2013. Thiết bị này ngay lập tức gây chú ý nhờ thiết kế nhỏ gọn, hiện đại và khả năng theo dõi chuyển động tay trong không gian 3D với độ chính xác cao (độ sai lệch dưới 0,01 mm). Phiên bản đầu tiên bao gồm:

* Phần cứng: Một thiết bị nhỏ gọn với khung nhôm nguyên khối, bề mặt kính cường lực chống xước. Bên trong tích hợp hai camera hồng ngoại và ba đèn LED hồng ngoại để theo dõi chuyển động tay.
* Phần mềm: SDK ban đầu hỗ trợ các hệ điều hành phổ biến như Windows và macOS, cung cấp khả năng phát hiện ngón tay, lòng bàn tay và cử chỉ cơ bản như nhấn, vuốt, nắm, và chỉ.

Tuy nhiên, hạn chế lớn của phiên bản đầu tiên là góc theo dõi khá hạn chế (chỉ khoảng 150 độ) và yêu cầu đặt thiết bị cố định trên mặt bàn, dẫn đến trải nghiệm không hoàn toàn tự nhiên khi kết hợp với VR/AR.

Hands reaching out to touch a computer screen

Description automatically generated

Vào năm 2016, Leap Motion ra mắt phiên bản phần mềm Leap Motion Controller 2, cùng với cải tiến phần cứng nhằm mở rộng khả năng theo dõi chuyển động:

* Phạm vi theo dõi tăng: Tầm hoạt động được mở rộng để theo dõi bàn tay và ngón tay ngay cả khi chúng không trực tiếp đối diện với thiết bị.
* Cải tiến phần mềm: SDK v2 hỗ trợ theo dõi chính xác hơn, bao gồm cả chuyển động phức tạp như gập ngón tay, cử động cổ tay và tương tác hai tay cùng lúc.
* Tích hợp với VR/AR: Hỗ trợ các hệ thống VR/AR thông qua phụ kiện gắn thiết bị lên kính VR như Oculus Rift và HTC Vive.

Phiên bản nâng cấp đã giải quyết nhiều hạn chế trước đó, đồng thời mở rộng ứng dụng vào thực tế ảo (VR) và thực tế tăng cường (AR), làm cho Leap Motion trở thành thiết bị phổ biến trong cộng đồng phát triển VR/AR.

A close-up of a virtual reality headset

Description automatically generated

Năm 2019, Leap Motion sáp nhập với Ultrahaptics, một công ty công nghệ tập trung vào cảm giác xúc giác siêu âm. Sau sáp nhập, Leap Motion đổi tên thành Ultraleap, tập trung vào việc phát triển công nghệ theo dõi bàn tay và trải nghiệm xúc giác không chạm.

Cũng trong giai đoạn này, Ultraleap giới thiệu phiên bản nâng cấp của thiết bị, thường được gọi là Leap Motion Controller 2 hoặc Ultraleap Hand Tracking Module. Thiết bị này mang lại nhiều cải tiến so với phiên bản đầu tiên:

* Tăng góc nhìn: Góc theo dõi mở rộng lên tới 180 độ theo chiều ngang và 170 độ theo chiều dọc, hỗ trợ trải nghiệm tự nhiên hơn.
* Phạm vi hoạt động lớn hơn: Khoảng cách theo dõi tăng lên tới 1 mét, so với giới hạn 60 cm của phiên bản đầu.
* Độ chính xác và tốc độ cao hơn: Độ trễ thấp hơn, nhận diện chính xác hơn cả trong các chuyển động phức tạp.
* Tích hợp chuyên sâu: Thiết kế dành cho các thiết bị VR/AR như kính Varjo XR-3 hoặc hệ thống công nghiệp.

A black rectangular object with a black oval

Description automatically generated

Sau Leap Motion Controller 2, Ultraleap tiếp tục mở rộng các dòng sản phẩm:

* Ultraleap Stereo IR 170 Module: Thiết bị theo dõi bàn tay tích hợp, tối ưu hóa cho các thiết bị VR/AR công nghiệp.
* TouchFree Application: Giải pháp không chạm dành cho màn hình cảm ứng, được ứng dụng trong y tế, bán lẻ, và các hệ thống công cộng.
* Haptics Technology: Kết hợp công nghệ theo dõi tay và cảm giác xúc giác không chạm trong không gian 3D.

Leap Motion Controller đã chứng minh khả năng tạo ra sự thay đổi lớn trong cách con người tương tác với công nghệ, và các phiên bản tiếp theo đang tiếp tục mở ra những tiềm năng ứng dụng mới.

* + 1. **Các ứng dụng**

Tính từ phiên bản đầu tiên, Leap Motion Controller và các dòng sản phẩm kế thừa đã có mặt trong nhiều lĩnh vực, từ giải trí đến y tế và giáo dục.

Leap Motion Controller đã trở thành một phần không thể thiếu trong lĩnh vực VR và AR, nhờ khả năng theo dõi tay chính xác và tự nhiên. Các ứng dụng tiêu biểu gồm:

* Leap Motion Interaction Engine: Một công cụ giúp tích hợp thiết bị vào môi trường VR/AR, hỗ trợ người dùng tương tác tự nhiên với các đối tượng ảo như kéo, thả, cầm nắm hoặc xoay vật thể.
* Varjo XR-3: Một trong những kính VR cao cấp nhất, tích hợp Leap Motion Controller để hỗ trợ theo dõi bàn tay mà không cần sử dụng bộ điều khiển (controller). Điều này cho phép các nhà thiết kế công nghiệp và kỹ sư dễ dàng thao tác trực tiếp với các mô hình 3D.
* STRATOS Inspire: Một giải pháp kết hợp công nghệ Leap Motion với cảm giác xúc giác không chạm, được ứng dụng trong các trải nghiệm AR tương tác tại triển lãm hoặc quảng cáo.

Leap Motion Controller đã được áp dụng trong lĩnh vực y tế, đặc biệt là trong các bài tập phục hồi chức năng và đào tạo:

* NeuroRehab VR: Một nền tảng VR y tế, nơi Leap Motion Controller được sử dụng để theo dõi và phân tích các chuyển động của tay, hỗ trợ bệnh nhân sau đột quỵ hoặc chấn thương phục hồi khả năng vận động.
* Hand Therapy Applications: Một số ứng dụng tập trung vào các bài tập vật lý trị liệu bằng cách yêu cầu người dùng thực hiện các cử chỉ tay trong không gian ảo. Điều này vừa tăng tính hiệu quả, vừa mang lại cảm giác thú vị hơn so với các phương pháp truyền thống.



Leap Motion Controller giúp tạo ra những bài học và chương trình đào tạo tương tác, cho phép học viên trải nghiệm thực tế hơn:

* Virtual Science Lab: Một ứng dụng VR nơi học sinh có thể sử dụng tay để thao tác trong phòng thí nghiệm ảo, chẳng hạn như trộn hóa chất, điều chỉnh kính hiển vi hoặc thực hiện các thí nghiệm vật lý.
* Flight Simulator Training: Trong các chương trình mô phỏng huấn luyện phi công, Leap Motion Controller được sử dụng để kiểm soát các công tắc và thao tác trong buồng lái, giúp tăng cường kỹ năng mà không cần mô hình vật lý thực.

A cockpit of a plane

Description automatically generated

Và không thể không kể dến lĩnh vực giải trí, Leap Motion Controller đã mang lại trải nghiệm độc đáo cho người dùng khi được tích hợp trong một số sản phẩm nổi bật:

* Hover Junkers: Một trò chơi bắn súng VR, trong đó người chơi sử dụng tay để bắn súng và điều khiển các đối tượng. Leap Motion Controller đóng vai trò theo dõi chính xác các động tác tay trong môi trường ảo.
* The Unspoken: Một tựa game VR nhập vai hành động, nơi người chơi hóa thân thành một pháp sư, sử dụng tay để thực hiện các phép thuật thông qua cử chỉ. Trò chơi này được phát triển bởi Insomniac Games và tích hợp hoàn hảo Leap Motion Controller, mang lại cảm giác chân thực khi thực hiện các thao tác.

A hand with a glowing blue light coming out of it

Description automatically generated

Như vậy, công nghệ nhận diện cử chỉ tay, với đại diện tiêu biểu là Leap Motion Controller, đã chứng minh tiềm năng to lớn trong việc định hình lại trải nghiệm người dùng, đặc biệt trong lĩnh vực giải trí và phát triển game. Khả năng theo dõi chuyển động chính xác và tự nhiên mở ra cơ hội sáng tạo không giới hạn cho các nhà phát triển, từ việc tạo ra các tựa game nhập vai độc đáo đến việc ứng dụng trong thực tế ảo (VR) và thực tế tăng cường (AR). Đây là nền tảng để các trò chơi tương lai không chỉ là giải trí mà còn mang tính tương tác vượt bậc, gắn kết người chơi với không gian kỹ thuật số theo những cách hoàn toàn mới mẻ.

1. **GIỚI THIỆU UNITY ENGINE VÀ ULTRALEAP PLUGIN**
   1. **Giới thiệu về Unity Engine**
      1. **Tổng quan**

Unity được biết đến như là một công cụ trò chơi đa nền tảng, nó được phát triển bởi Unity Technologies. Mục đích sử dụng chủ yếu là để phát triển trò chơi điện tử và mô phỏng cho máy tính, thiết bị di động, bảng điều khiển,…Nhờ vào tính năng đa nền tảng, Unity là cái tên phổ biến với cả các nhà phát triển game tự do cũng như trong các studio game. Nó được dùng nhằm tạo ra những trò chơi như Hearthstone, Cuphead, Pokemon Go, Rimworld cùng vô vàn trò chơi khác nữa.

A black and white image of a black box

Description automatically generated with medium confidence

*Hình 1.3.1. Logo Unity*

Các sản phẩm của Unity 2D, 3D có thể được lập trình dựa trên 3 ngôn ngữ là C#, JavaScript và Boo. Trong đó, C# là ngôn ngữ chính mà rất nhiều lập trình viên Unity sử dụng cho tới thời điểm hiện tại, JavaScript và Boo đã không còn được hỗ trợ do ít người dùng và khó bảo trì. Xuất hiện cách đây khá lâu từ năm 2005, Unity đã có một lượng lớn người dùng cũng như sở hữu một thư viện tài nguyên khổng lồ. Không chỉ có tài liệu tuyệt vời, mà Unity còn có vô vàn video cùng những hướng dẫn trực tuyến đáng ngạc nhiên dành cho người dùng. Chính vì lý do này, Unity trở thành sự lựa chọn vô cùng sáng suốt cho người mới bước đầu tiếp cận với các công cụ game. Có mặt trong danh sách những công cụ trò chơi điện tử, Unity giữ vai trò tựa như một cổng thông tin tài nguyên và kiến thức, được xây dựng, phát triển chỉ dựa trên cộng đồng rộng lớn của riêng họ.

Unity là một trong những Game Engine phổ biến và dễ dùng, sự phát triển của Unity Engine theo thời gian cũng đem lại nhiều tính năng mới và hỗ trợ cho các nền tảng đa dạng:

* Unity 1.0 (June 2005): Hỗ trợ phát triển trò chơi cho Mac OS X, sử dụng ngôn ngữ JavaScript cho việc lập trình.
* Unity 2.0 (November 2007): Hỗ trợ phát triển trò chơi cho Windows, cải thiện đồ họa và hiệu năng.
* Unity 3.0 (October 2010): Hỗ trợ IOS, cung cấp cửa sổ Scene và Game để dễ dàng xem kịch bản trò chơi và trò chơi thực tế.
* Unity 4.0 (November 2012): Hỗ trợ Android và Adobe Flash, Cải thiện đa nền tảng, với khả năng phát triển trò chơi cho nhiều nền tảng dễ dàng hơn.
* Unity 5.0 (March 2015): Ánh sáng thực giác với Global Illumination, tích hợp trình quản lý Assets, cải thiện hệ thống lực cắt và API đồ họa.
* Unity 2017.1 (July 2017): Tích hợp Timeline cho quản lý trình tự sự kiện, cải thiện đa nền tảng và hiệu năng trên nhiều nền tảng.
* Unity 2018.1 (April 2018): Scriptable Render Pipelines cho kiểm soát cao hơn về đồ họa, cải thiện công cụ đồ họa và lập trình.
* Unity 2019.1 (April 2019): DOTS (Data-Oriented Technology Stack) giúp tối ưu hóa hiệu năng, cải thiện hệ thống gắn kết và lập trình AI.
* Unity 2020.1 (May 2020): Visual Effect Graph cho hiệu ứng đồ họa nâng cao, hỗ trợ WebGL.
* Unity 2021.1 (April 2021): Hỗ trợ bất đồng bộ (Async) và C# 8.0, Universal Render Pipeline.
* Unity 2022.1 (March 2022): Tiếp tục cải thiện hiệu năng và đa nền tảng.
* Ngoài ra Unity cũng có các phiên bản Long Term Support (LTS) cho người dùng muốn duy trì và phát triển game/content trên một phiên bản ổn định trong thời gian dài, Các phiên bản LTS bao gồm Unity2022.x, 2023.x, …
  + 1. **Mục đích hình thành và phát triển**

Mục đích hình thành và phát triển của Unity là tạo ra một môi trường phát triển trò chơi mạnh mẽ, đa nền tảng, và dễ sử dụng. Unity được thiết kế để giúp các nhà phát triển tạo ra trò chơi 2D và 3D chất lượng cao trên nhiều nền tảng khác nhau một cách hiệu quả. Dưới đây là một số điểm cải tiến của Unity:

* Đa nền tảng: Unity cho phép phát triển trò chơi cho nhiều nền tảng, bao gồm PC, Mac, điện thoại di động, máy tính bảng, cả kính thực tế ảo và thực tế ảo. Điều này giúp giảm thời gian và công sức phát triển trò chơi trên nhiều nền tảng.
* Dễ sử dụng: Unity được thiết kế với giao diện người dùng trực quan, giúp người dùng dễ dàng tạo, chỉnh sửa và quản lý tài liệu, tài nguyên, và kịch bản trong trò chơi.
* Cộng đồng lớn: Unity có một cộng đồng phát triển lớn, với hàng triệu nhà phát triển trên khắp thế giới. Điều này tạo điều kiện tốt cho việc học hỏi, hỗ trợ và chia sẻ tài liệu và tài nguyên.
* Đồ họa và âm thanh: Unity cung cấp công cụ mạnh mẽ cho đồ họa 2D và 3D, với sự hỗ trợ của các công cụ và tích hợp sẵn cho hiệu ứng đặc biệt, ánh sáng, và âm thanh.
* Hiệu suất và tối ưu hóa: Unity hỗ trợ tối ưu hóa hiệu suất và làm việc trên các thiết bị di động và nền tảng khác nhau. Điều này giúp trò chơi chạy mượt mà trên nhiều loại thiết bị.
* Scripting đa ngôn ngữ: Unity hỗ trợ nhiều ngôn ngữ lập trình như C#, JavaScript, và Boo, cho phép nhà phát triển lựa chọn ngôn ngữ ưa thích của họ.
* Hỗ trợ mạnh mẽ: Unity cung cấp tài liệu phong phú, hướng dẫn, video học, và hỗ trợ kỹ thuật để giúp người dùng giải quyết các vấn đề trong quá trình phát triển trò chơi.
* Khả năng mở rộng và tùy chỉnh: Unity cho phép người dùng tạo ra các plug-in và mở rộng để tùy chỉnh môi trường phát triển của họ.

Như vậy, mục đích chính của Unity là tạo điều kiện thuận lợi cho các nhà phát triển để tạo ra trò chơi chất lượng cao, đa nền tảng, và mang tính thương mại. Unity giúp giảm thời gian và công sức phát triển, giúp tập trung vào việc sáng tạo và xây dựng trải nghiệm trò chơi độc đáo.

* + 1. **Ưu và nhược điểm của Unity**

Ưu điểm của Unity:

* Trên Editor, các nhà phát triển game không nhất thiết phải viết code nhằm sắp xếp những đối tượng trong game như các Engine khác. Thay vào đó, các Developer có thể kéo thả hoặc thay đổi vị trí của từng đối tượng trong game một cách trực tiếp.
* Có thể sử dụng đa nền tảng: Đây là một ưu điểm quan trọng vì nó giúp tiết kiệm nhiều công sức và chi phí cho doanh nghiệp. Hãy hình dung việc bạn tạo ra một chương trình game mà nó có thể chạy được trên hầu hết hệ điều hành như Mobile (iOS, Android), Desktop (Window, Mac và Linux) hoặc Web (WebGL).
* Sử dụng miễn phí, đây là ưu điểm thu hút đông đảo Developer chọn làm việc với game engine này. Song, với việc game được tạo ra miễn phí thì điều bắt buộc là phải có Logo Unity trong game đó.
* Unity hỗ trợ rất nhiều định dạng asset khác nhau và có thể tự động di chuyển đến định dạng phù hợp nhất với nền tảng thích hợp.
* Khả năng dùng phổ biến C#
* Có công cụ rất trực quan, editor có thể mở rộng bằng plugins

Nhược điểm của Unity:

* Dung lượng Unity game bundle khá lớn: Khi đặt lên bàn cân với những game engine khác, game mà Unity sản xuất có dung lượng nặng và đây là điểm trừ lớn. Theo giới chuyên gia, một game tốt chỉ nên nặng dưới 100MB, tuy nhiên game web do Unity sản xuất thường có dung lượng lên đến cả trăm MB nên việc giật, lag là khó tránh khỏi.
* Unity thường cho ra mắt nhiều bản cập nhật trong một năm và liên tục trong nhiều năm. Trong trường hợp bản cập nhật khác năm, trong khuôn khổ nội bộ một team hoặc giữa các team phát triển và khách hàng, việc sử dụng nhiều phiên bản Unity khác năm với nhau thì khi đẩy code lên dễ gây ra tình trạng lỗi hình ảnh, code, script… trường hợp xấu hơn là mất hết tiến độ.
* Việc hợp tác được xem là khá khó khăn. Unity sử dụng một server asset rất hiệu quả để giúp cho các đội phát triển phần mềm dễ dàng hợp tác với nhau. Tuy nhiên, nếu không sử dụng unity thì việc chia sẻ asset và code giữa các thành viên trong team có thể gây ra những vấn đề nghiêm trọng. Lựa chọn tốt nhất là sử dụng một số công cụ quản lý resource bên ngoài nhưng có một vài binary file không thể merge được với nhau và việc cập nhật asset có thể gây nên một số vấn đề trong scenes, mất kết nối đến script và các đối tượng khác
* Không thích hợp với những dự án lớn: không thể đào sâu quá vừa được xem là nhược điểm cũng vừa được xem là ưu điểm. Một mặt, unity cho phép quy trình nhanh chóng, thích ứng với những người mới bắt đầu, mặt khác điều đó được xem như unity không phải là thứ bạn tìm kiếm nếu bạn đang hy vọng sẽ tạo ra thứ gì đó thật đặc biệt, khác lạ hay trong một quy mô lớn.
* Mã nguồn của engine không được công bố dù cho những người dùng chấp nhận chi trả tiền. Điều đó có nghĩa là nếu bạn gặp một bug với engine thì lúc này bạn phải chờ unity fix chúng trong các bản tiếp theo. Từ đó có thể gây ra những vấn đề nghiêm trọng với project của bạn
  + 1. **Một số thành phần chính trong Unity Editor**

**Cửa sổ Scene**: Phần này phần hiển thị các đối tượng trong scene một cách trực quan, có thể lựa chọn các đối tượng, kéo thả, phóng to, thu nhỏ, xoay các đối tượng ... Phần này có để thiết lập một số thông số như hiển thị ánh sáng, âm anh, cách nhìn 2D hay 3D... Khung nhìn Scene là nơi bố trí các GameObject như cây cối, cảnh quan, enemy, player, camera, … trong game. Sự bố trí hoạt cảnh là một trong những chức năng quan trọng nhất của Unity.

A screenshot of a computer

Description automatically generated

*Hình 1.3.2. Cửa sổ Scene*

**Cửa sổ Hierarchy**: Tab hierarchy là nơi hiển thị các Gameobject trong Scenes hiện hành. Khi các đối tượng được thêm hoặc xóa trong Scenes, tương ứng với các đối tượng đó trong cửa sổ Hierarchy. Bạn có thể tạo đối tượng mới bằng cách nhấp chuột phải vào cửa sổ Hierarchy và chọn "Create Empty" hoặc chọn đối tượng trong Scene và sao chép nó. Có thể xóa đối tượng bằng cách nhấp chuột phải và chọn "Delete" hoặc kéo nó ra khỏi cửa sổ Hierarchy. Bạn cũng có thể sắp xếp lại đối tượng bằng cách kéo và thả chúng trong cây. Bạn có thể lọc đối tượng bằng cách sử dụng thanh tìm kiếm. Có thể chỉnh sửa tên của đối tượng bằng cách nhấn đúp vào tên nó trong Hierarchy.Cửa sổ Hierarchy là một công cụ quan trọng giúp bạn quản lý và tương tác với tất cả các đối tượng trong Scene của mình trong Unity.

A screenshot of a computer

Description automatically generated

*Hình 1.3.3. Cửa sổ Hierarchy trong Unity*

**Cửa sổ Game**: Đây là màn hình demo Game, là góc nhìn từ camera trong game. Thanh công cụ trong cửa sổ game cung cấp các tùy chỉnh về độ phân giải màn hình, thông số (stats), gizmos, tùy chọn bật tắt các component…

A screenshot of a computer

Description automatically generated

*Hình 1.3.4. Cửa sổ Game trong Unity*

**Cửa sổ Project**: Đây là cửa sổ explorer của Unity, hiển thị thông tin của tất cả các tài nguyên (Assets) trong game của bạn. Cột bên trái hiển thị assets và các mục yêu thích dưới dạng cây thư mục tương tự như Windows Explorer. Khi click vào một nhánh trên cây thư mục thì toàn bộ nội dung của nhánh đó sẽ được hiển thị ở khung bên phải. Ta có thể tạo ra các thư mục mới bằng cách Right click -> Create -> Folder hoặc nhấn vào nút Create ở góc trên bên trái cửa sổ Project và chọn Folder. Các tài nguyên trong game cũng có thể được tạo ra bằng cách này. Phía trên cây thư mục là mục Favorites, giúp chúng ta truy cập nhanh vào những tài nguyên thường sử dụng. Chúng ta có thể đưa các tài nguyên vào Favorites bằng thao tác kéo thả. Đường dẫn của thư mục tài nguyên hiện tại. Chúng ta có thể dễ dàng tiếp cận các thư mục con hoặc thư mục gốc bằng cách click chuột vào mũi tên hoặc tên thư mục.

A screenshot of a computer

Description automatically generated

*Hình 1.3.5. Cửa sổ Project trong Unity*

**Cửa sổ Inspector**: Cửa sổ Inspector hiển thị chi tiết các thông tin về Gameobject đang làm việc, kể cả những component được đính kèm và thuộc tính của nó. Bạn có thể điều chỉnh, thiết lập mọi thông số và chức năng của Gameobject thông qua cửa sổ Inspector.Mọi thuộc tính thể hiện trong Inspector đều có thể dễ dàng tùy chỉnh trực tiếp mà không cần thông qua một kịch bản định trước. Tuy nhiên Scripting API cung cấp một số lượng nhiều và đầy đủ hơn do giao diện Inspector là có giới hạn. Các thiết lập của từng component được đặt trong menu. Các bạn có thể click chuột phải, hoặc chọn icon hình bánh răng nhỏ để xuất hiện menu. Ngoài ra Inspector cũng thể hiện mọi thông số Import Setting của asset đang làm việc như hiển thị mã nguồn của Script, các thông số animation, …

A screenshot of a computer

Description automatically generated

*Hình 1.3.6. Cửa sổ Inspector trong Unity*

* + 1. **Một số khái niệm trong Unity Editor**

**GameObject**: Một đối tượng cụ thể trong game gọi là một Gameobject, có thể là nhân vật, đồ vật nào đó. Ví dụ: cây cối, xe cộ, nhà cửa, người…

**Component**: Một GameObject sẽ có nhiều thành phần cấu tạo nên nó như là hình ảnh (sprite render), tập hợp các hành động (animator), thành phần xử lý va chạm (collision), tính toán vật lý (physical), mã điều khiển (script), các thành phần khác... mỗi thứ như vậy gọi là một component của GameObject.

**Sprite**: Là một hình ảnh 2D của một gameobject có thể là hình ảnh đầy đủ, hoặc có thể là một bộ phận nào đó.

**Animation**: Là tập một hình ảnh động dựa trên sự thay đổi liên tục của nhiều sprite khác nhau.

**Prefabs**: Là một khái niệm trong Unity, dùng để sử dụng lại các đối tượng giống nhau có trong game mà chỉ cần khởi tạo lại các giá trị vị trí, tỉ lệ biến dạng và góc quay từ một đối tượng ban đầu. Ví dụ: Các đối tượng là đồng tiền trong game Mario đều có xử lý giống nhau, nên ta chỉ việc tạo ra một đối tượng ban đầu, các đồng tiền còn lại sẽ sử dụng prefabs. Hoặc khi ta lát gạch cho một cái nền nhà, các viên gạch cũng được sử dụng là prefabs.

**Sound**: Âm thanh trong game.

**Script**: Script là tập tin chứa các đoạn mã nguồn, dùng để khởi tạo và xử lý các đối tượng trong game. Trong Unity có thể dùng C#, Javascript để lập trình Script.

**Scenes**: Quản lý tất cả các đối tượng trong một màn chơi của game.

**Assets**: Bao gồm tất cả những gì phục vụ cho dự án game như sprite, animation, sound, script, scenes…

**Camera**: Là một gameobject đặc biệt trong scene, dùng để xác định tầm nhìn, quan sát các đối tượng khác trong game.

**Transform**: Là 3 phép biến đổi tịnh tiến, quay theo các trục, và phóng to thu nhỏ một đối tượng

* 1. Giới thiệu về Ultraleap plugin
     1. Tổng quan về Ultraleap plugin
     2. Các thành phần chính trong Ultraleap
  2. Tích hợp Ultraleap vào Unity
     1. Chuẩn bị môi trường làm việc
     2. Cài đặt Ultraleap plugin
     3. Các ứng dụng của Ultraleap trong phát triển game
  3. Các vấn đề thường gặp và cách giải quyết

1. **PHÂN TÍCH VÀ THIÊT KẾ GAME CHÉM HOA QUẢ**
   1. Biểu đồ phân rã chức năng
   2. Biểu đồ usecase
   3. Kịch bản usecase
   4. Biểu đồ lớp
   5. Biểu đồ tuần tự
2. **XÂY DỰNG GAME CHÉM HOA QUẢ CÓ TÍCH HỢP LEAP MOTION VỚI UNITY ENGINE**
   1. Xây dựng hệ thống tạo hoa quả
   2. Tích hợp Ultraleap
   3. Xử lý logic game
   4. Cài đặt giao diện cơ bản
   5. 4 hoá và kiểm thử
3. **TÀI LIỆU THAM KHẢO**

Leap Motion paper: <https://www.scitepress.org/Papers/2021/109507/109507.pdf>

Performance: <https://www.mdpi.com/1424-8220/22/13/4880>

Thông số: [thông số](https://www.ultraleap.com/datasheets/Leap_Motion_Controller_Datasheet.pdf?_gl=1*aq5w96*_ga*MTM2MDY1MDE2Ni4xNzI4Mjg0OTcz*_ga_5G8B19JLWG*MTczNDI1ODA4Ny4xNi4xLjE3MzQyNTgxMzcuMTAuMC4w)

<https://www.slideshare.net/slideshow/leap-motion-controller-and-application-development/77979588>

video demo con mèo: <https://www.ultraleap.com/tracking/>