

# BÁO CÁO DỰ ÁN SUICHIN:

## ỨNG DỤNG BLOCKCHAIN SUI TRONG PHÁT TRIỂN TRÒ CHƠI WEB3

**Đơn vị thực hiện:** Học viện Công nghệ Bưu chính Viễn thông (PTIT)

**Đội nghiên cứu:** Nhà thám hiểm Ixtal

**Chương trình:** Sui Campus Hackathon

**Thời gian thực hiện:** Tháng 02/2026

## TÓM TẮT

Báo cáo này trình bày chi tiết quá trình nghiên cứu, thiết kế và triển khai dự án SuiChin, một ứng dụng trò chơi Web3 được xây dựng trên nền tảng Sui Blockchain. Dự án được thực hiện trong khuôn khổ Sui Campus Hackathon với mục tiêu khám phá tiềm năng ứng dụng công nghệ blockchain vào lĩnh vực game và giải trí số.

SuiChin kết hợp gameplay truyền thống dựa trên cơ chế vật lý với các tính năng đặc trưng của Web3, bao gồm quản lý tài sản số phi tập trung, cơ chế NFT (Non-Fungible Token) và hệ thống thành tích on-chain. Hệ thống được thiết kế theo mô hình hybrid, trong đó logic gameplay được xử lý off-chain nhằm đảm bảo hiệu năng và trải nghiệm người dùng, trong khi các dữ liệu quan trọng như hồ sơ người chơi, kết quả trận đấu và tài sản số được lưu trữ bất biến trên blockchain thông qua smart contract viết bằng ngôn ngữ Move.

Nghiên cứu này đóng góp vào việc làm rõ cách thức cân bằng giữa yêu cầu về hiệu năng thời gian thực của game và tính minh bạch, bảo mật vốn có của công nghệ blockchain, đồng thời minh họa khả năng ứng dụng thực tiễn của Sui Blockchain trong việc xây dựng các ứng dụng phi tập trung có tính tương tác cao.

## 1. GIỚI THIỆU

### 1.1. Bối cảnh nghiên cứu

Sự phát triển của công nghệ blockchain trong thập kỷ qua đã mở ra nhiều hướng ứng dụng mới ngoài lĩnh vực tài chính truyền thống. Đặc biệt, khái niệm Web3 và GameFi (Game Finance) đã nổi lên như một xu hướng đổi mới trong ngành công nghiệp game, hứa hẹn mang lại quyền sở hữu thực sự và giá trị kinh tế cho người chơi thông qua các cơ chế phi tập trung.

Sui Blockchain, được phát triển bởi Myster Labs, là một nền tảng blockchain thế hệ mới được thiết kế đặc biệt để hỗ trợ các ứng dụng yêu cầu thông lượng cao và độ trễ thấp. Với kiến trúc dựa trên ngôn

ngữ lập trình Move và cơ chế xử lý giao dịch song song, Sui hứa hẹn khắc phục các hạn chế về hiệu năng của các nền tảng blockchain trước đó.

## 1.2. Vấn đề nghiên cứu

Việc tích hợp blockchain vào game đặt ra nhiều thách thức kỹ thuật, bao gồm:

- Đảm bảo hiệu năng thời gian thực cho gameplay trong khi duy trì tính minh bạch và bảo mật của blockchain
- Thiết kế cơ chế kinh tế trong game cân bằng giữa tính bền vững và hấp dẫn người chơi
- Quản lý tài sản số một cách an toàn và hiệu quả
- Ngăn chặn các hành vi gian lận và tấn công trong môi trường phi tập trung

## 1.3. Mục tiêu nghiên cứu

Dự án SuiChin hướng đến các mục tiêu cụ thể sau:

- Xây dựng một ứng dụng game Web3 đầy đủ chức năng trên nền tảng Sui Blockchain, minh chứng khả năng ứng dụng thực tế của công nghệ
- Thiết kế và triển khai kiến trúc hybrid off-chain/on-chain nhằm tối ưu hóa cả hiệu năng lẫn tính bảo mật
- Phát triển hệ thống smart contract quản lý hồ sơ người chơi, tài sản số và thành tích một cách minh bạch
- Đánh giá khả năng mở rộng và tính khả thi của giải pháp trong điều kiện thực tế

## 1.4. Phạm vi nghiên cứu

Do thời gian thực hiện trong khuôn khổ hackathon có hạn, nghiên cứu tập trung vào các khía cạnh cốt lõi:

- Gameplay: Trò chơi bùng chun 2D với chế độ chơi đơn (người chơi đấu với bot)
- Quản lý dữ liệu: Hồ sơ người chơi được lưu trữ và quản lý hoàn toàn on-chain
- Tài sản số: Hai loại NFT bao gồm vật phẩm có thể chuyển nhượng và thành tích không thể chuyển nhượng
- Môi trường triển khai: Sui Testnet (chưa triển khai lên mainnet)
- Giới hạn: Chưa triển khai chế độ multiplayer thời gian thực, chưa tích hợp cơ chế tokenomics phức tạp

## 1.5. Ý nghĩa nghiên cứu

Dự án SuiChin đóng góp vào lĩnh vực nghiên cứu Web3 và GameFi thông qua:

- Cung cấp mô hình tham chiếu cho việc xây dựng game Web3 trên Sui Blockchain
- Minh chứng tính khả thi của kiến trúc hybrid trong việc cân bằng hiệu năng và tính phi tập trung
- Đề xuất các giải pháp kỹ thuật cụ thể cho việc quản lý trạng thái game và tài sản số

- Tạo nền tảng cho các nghiên cứu mở rộng về game blockchain và ứng dụng phi tập trung

## 2. CƠ SỞ LÝ THUYẾT VÀ CÔNG NGHỆ

### 2.1. Công nghệ Blockchain và Web3

Blockchain là công nghệ số cái phân tán cho phép lưu trữ dữ liệu theo cách phi tập trung, bất biến và minh bạch. Mỗi khối dữ liệu được liên kết mật mã với khối trước đó, tạo thành chuỗi khối không thể thay đổi hồi tố mà không bị phát hiện.

Web3 đại diện cho thế hệ tiếp theo của internet, trong đó người dùng có quyền sở hữu và kiểm soát hoàn toàn dữ liệu và tài sản số của mình thông qua các giao thức phi tập trung. Khác với Web2 tập trung vào các nền tảng do doanh nghiệp kiểm soát, Web3 trao quyền cho người dùng thông qua ví mật mã và smart contract.

### 2.2. Sui Blockchain

Sui là nền tảng blockchain layer-1 được thiết kế để hỗ trợ các ứng dụng yêu cầu thông lượng cao và độ trễ thấp. Các đặc điểm kỹ thuật nổi bật của Sui bao gồm:

- **Kiến trúc đối tượng:** Sui sử dụng mô hình dữ liệu theo hướng đối tượng, trong đó mỗi đối tượng có chủ sở hữu rõ ràng và có thể được xử lý độc lập
- **Xử lý song song:** Các giao dịch không có xung đột về tài nguyên có thể được xử lý song song, tăng đáng kể thông lượng hệ thống
- **Consensus Byzantine Fault Tolerant:** Sui sử dụng thuật toán đồng thuận Narwhal-Bullshark, tối ưu hóa cho tốc độ finality nhanh
- **Ngôn ngữ Move:** Smart contract trên Sui được viết bằng Move, ngôn ngữ lập trình được thiết kế đặc biệt cho blockchain với các cơ chế bảo mật tích hợp

### 2.3. Ngôn ngữ lập trình Move

Move là ngôn ngữ lập trình được phát triển ban đầu cho dự án Libra/Diem của Meta, sau đó được áp dụng và phát triển thêm bởi Sui và Aptos. Các đặc điểm quan trọng của Move:

- **Resource-oriented programming:** Move định nghĩa khái niệm "resource" - loại dữ liệu đặc biệt không thể sao chép hoặc hủy tùy ý, phù hợp cho việc biểu diễn tài sản số
- **Type safety:** Hệ thống kiểu mạnh của Move giúp phát hiện lỗi tại thời điểm biên dịch, giảm thiểu rủi ro bảo mật
- **Ownership model:** Move áp dụng mô hình ownership rõ ràng, đảm bảo mỗi tài sản luôn có chủ sở hữu duy nhất tại một thời điểm
- **Module system:** Cấu trúc module hóa giúp tổ chức code rõ ràng và tái sử dụng

### 2.4. NFT và tài sản số trên Blockchain

NFT (Non-Fungible Token) là token không thể thay thế, đại diện cho quyền sở hữu duy nhất đối với một tài sản số hoặc vật lý. Trên Sui, NFT được triển khai thông qua các đối tượng có thuộc tính key ability, cho phép chúng được lưu trữ độc lập và chuyển giao giữa các địa chỉ.

Trong bối cảnh game, NFT có thể đại diện cho:

- Vật phẩm trong game: vũ khí, trang bị, skin có thể giao dịch tự do
- Nhân vật hoặc thú cưng: có thuộc tính và cấp độ độc đáo
- Thành tích: bằng chứng về các mốc quan trọng người chơi đạt được
- Đất đai ảo: không gian trong thế giới game có thể sở hữu

## 2.5. Kiến trúc Hybrid Off-chain/On-chain

Kiến trúc hybrid là giải pháp cân bằng giữa hiệu năng và tính phi tập trung bằng cách phân chia xử lý giữa môi trường off-chain và on-chain:

- **Off-chain:** Xử lý các tác vụ yêu cầu tốc độ cao như render đồ họa, physics engine, AI đối thủ, và các tương tác người dùng tức thời
- **On-chain:** Lưu trữ các dữ liệu quan trọng cần tính bất biến và minh bạch như hồ sơ người chơi, kết quả trận đấu, tài sản số và giao dịch

Cách tiếp cận này cho phép game duy trì trải nghiệm mượt mà tương đương game truyền thống trong khi vẫn tận dụng được lợi ích của blockchain về quyền sở hữu tài sản và tính minh bạch.

# 3. PHÂN TÍCH YÊU CẦU HỆ THỐNG

## 3.1. Yêu cầu chức năng

Hệ thống SuiChin cần đáp ứng các yêu cầu chức năng cốt lõi sau:

### 3.1.1. Quản lý người chơi

- Tạo và quản lý hồ sơ người chơi được gắn với địa chỉ ví Sui duy nhất
- Lưu trữ thông tin trạng thái người chơi bao gồm số dư token trong game, chuỗi thắng liên tiếp, và các chỉ số thống kê
- Cho phép người chơi xem lịch sử hoạt động và thành tích của mình
- Đảm bảo mỗi địa chỉ ví chỉ có thể tạo một hồ sơ duy nhất

### 3.1.2. Gameplay và cơ chế chơi

- Triển khai gameplay bùng chun 2D với physics engine chính xác
- Xử lý va chạm giữa các đối tượng trong game theo quy luật vật lý
- Cung cấp đối thủ AI với độ khó hợp lý
- Xác định điều kiện thắng/thua rõ ràng
- Đảm bảo tốc độ khung hình ổn định (target 60 FPS)

### 3.1.3. Ghi nhận kết quả

- Gửi giao dịch lên blockchain sau mỗi ván chơi kết thúc
- Cập nhật hồ sơ người chơi với kết quả mới (thắng/thua, điểm số, streak)
- Tính toán và cộng/trừ token trong game dựa trên kết quả

- Tạo bản ghi bất biến về lịch sử trận đấu

### **3.1.4. Hệ thống kinh tế**

- Cung cấp cơ chế faucet để người chơi mới nhận token khởi đầu
- Quản lý số dư token của mỗi người chơi
- Cho phép chi tiêu token để mua vật phẩm hoặc tính năng đặc biệt
- Thưởng token khi người chơi đạt thành tích

### **3.1.5. NFT và tài sản số**

- Phát hành NFT vật phẩm có thể chuyển nhượng tự do giữa các người chơi
- Phát hành NFT thành tích gắn cố định với người chơi, không thể chuyển nhượng (soulbound)
- Hiển thị kho NFT của người chơi với metadata đầy đủ
- Hỗ trợ tích hợp với marketplace bên thứ ba để giao dịch NFT vật phẩm

## **3.2. Yêu cầu phi chức năng**

### **3.2.1. Hiệu năng**

- Gameplay mượt mà với độ trễ input dưới 50ms
- Thời gian phản hồi giao diện dưới 200ms
- Thời gian xác nhận giao dịch trên blockchain dưới 2 giây (trên testnet)
- Khả năng xử lý đồng thời nhiều người chơi mà không ảnh hưởng hiệu năng

### **3.2.2. Bảo mật**

- Dữ liệu quan trọng không thể bị thay đổi sau khi ghi lên blockchain
- Chỉ chủ sở hữu ví mới có thể cập nhật hồ sơ của mình
- Ngăn chặn các hành vi gian lận như thao túng kết quả trận đấu
- Smart contract phải vượt qua kiểm tra bảo mật cơ bản

### **3.2.3. Khả năng sử dụng**

- Giao diện trực quan, dễ hiểu cho người dùng không có kinh nghiệm blockchain
- Quy trình kết nối ví đơn giản với hướng dẫn rõ ràng
- Thông báo lỗi và hướng dẫn xử lý dễ hiểu
- Hỗ trợ responsive design cho các kích thước màn hình khác nhau

### **3.2.4. Khả năng mở rộng**

- Kiến trúc module hóa cho phép thêm tính năng mới dễ dàng
- Smart contract được thiết kế để có thể nâng cấp trong tương lai
- Cơ sở dữ liệu on-chain có cấu trúc linh hoạt

- Code được tài liệu hóa đầy đủ để đội ngũ khác có thể tiếp tục phát triển

### 3.2.5. Độ tin cậy

- Hệ thống ổn định, không bị crash trong quá trình chơi bình thường
- Xử lý lỗi graceful khi có sự cố mạng hoặc blockchain
- Dữ liệu người chơi được bảo toàn ngay cả khi ứng dụng gặp sự cố
- Có cơ chế phục hồi khi giao dịch blockchain thất bại

## 3.3. Ràng buộc kỹ thuật

- Triển khai trên Sui Testnet do giới hạn về chi phí và thời gian
- Sử dụng ngôn ngữ Move cho smart contract
- Frontend phải tương thích với Sui Wallet extension
- Tuân thủ các best practices của Sui ecosystem
- Hoàn thành trong khuôn khổ thời gian hackathon (khoảng 2-4 tuần)

# 4. THIẾT KẾ KIẾN TRÚC HỆ THỐNG

## 4.1. Tổng quan kiến trúc

SuiChin áp dụng kiến trúc hybrid ba tầng, phân tách rõ ràng giữa tầng giao diện, tầng xác thực và tầng lưu trữ dữ liệu. Kiến trúc này được thiết kế để tối ưu hóa cả trải nghiệm người dùng lẫn tính bảo mật và minh bạch của blockchain.

### Kiến trúc tổng thể hệ thống SuiChin

Các thành phần chính bao gồm:

- **Frontend Layer:** Ứng dụng web React xử lý giao diện người dùng, gameplay và render đồ họa
- **Wallet Layer:** Sui Wallet đóng vai trò trung gian xác thực danh tính và ký các giao dịch
- **Blockchain Layer:** Sui Blockchain lưu trữ smart contract và dữ liệu bất biến

Luồng dữ liệu trong hệ thống tuân theo nguyên tắc: mọi thay đổi trạng thái quan trọng đều phải được xác thực thông qua chữ ký số từ ví người dùng trước khi được ghi nhận lên blockchain.

## 4.2. Tầng Frontend và xử lý Gameplay

### 4.2.1. Công nghệ sử dụng

Frontend được xây dựng trên nền tảng công nghệ web hiện đại:

- **React 18:** Framework JavaScript cho xây dựng giao diện người dùng reactive
- **Sui SDK:** Thư viện JavaScript để tương tác với Sui Blockchain
- **Canvas API hoặc WebGL:** Render đồ họa 2D cho gameplay

- **Physics Engine:** Thư viện xử lý vật lý như Matter.js hoặc Planck.js

### 4.2.2. Game Loop

Vòng lặp game chính chạy ở tốc độ 60 FPS và thực hiện các bước sau mỗi frame:

```
function gameLoop(timestamp) {
// Tính delta time
const deltaTime = timestamp - lastFrameTime;
lastFrameTime = timestamp;

// Cập nhật vật lý
updatePhysics(deltaTime);

// Kiểm tra va chạm
checkCollisions();

// Cập nhật AI
updateAI(deltaTime);

// Kiểm tra điều kiện kết thúc
if (checkGameEnd()) {
handleGameEnd();
}

// Render scene
renderScene();

// Tiếp tục vòng lặp
requestAnimationFrame(gameLoop);
}
```

### 4.2.3. Xử lý Input

Hệ thống xử lý input từ người chơi bao gồm:

- Phát hiện và xử lý sự kiện chuột/touch để xác định hướng và lực búng
- Tính toán vector lực dựa trên khoảng cách và hướng kéo
- Áp dụng lực vào đối tượng chun thông qua physics engine
- Giới hạn lực tối đa để đảm bảo cân bằng gameplay

### 4.2.4. Quản lý trạng thái ứng dụng

Frontend sử dụng React Context API hoặc state management library để quản lý:

- Trạng thái kết nối ví (connected/disconnected)
- Thông tin người chơi hiện tại (address, balance, streak)
- Trạng thái game (menu, playing, paused, game over)
- Danh sách NFT của người chơi

## 4.3. Tầng Wallet và xác thực

### **4.3.1. Tích hợp Sui Wallet**

Ứng dụng tích hợp với Sui Wallet extension thông qua Wallet Standard API, cho phép:

- Phát hiện và kết nối với ví đã cài đặt
- Yêu cầu quyền truy cập vào địa chỉ ví
- Gửi yêu cầu ký giao dịch
- Lắng nghe các sự kiện thay đổi tài khoản

### **4.3.2. Luồng xác thực**

Quy trình xác thực người chơi tuân theo các bước:

1. Người chơi nhấn nút "Connect Wallet" trên giao diện
2. Frontend gửi yêu cầu kết nối đến Sui Wallet
3. Ví hiển thị popup xác nhận cho người dùng
4. Sau khi chấp thuận, ví trả về địa chỉ công khai
5. Frontend truy vấn blockchain để lấy hoặc tạo hồ sơ người chơi
6. Người chơi được xác thực và có thể bắt đầu chơi

### **4.3.3. Ký giao dịch**

Mọi giao dịch thay đổi trạng thái on-chain đều phải được ký bởi ví:

- Frontend xây dựng transaction block với các move call cần thiết
- Gửi transaction block chưa ký đến ví
- Ví hiển thị chi tiết giao dịch cho người dùng xem xét
- Người dùng xác nhận và ví ký giao dịch với private key
- Frontend gửi giao dịch đã ký lên blockchain
- Chờ transaction finality và cập nhật UI

## **4.4. Tầng Blockchain và Smart Contract**

### **4.4.1. Cấu trúc Module**

Smart contract được tổ chức thành các module Move độc lập:

- **player\_profile:** Quản lý hồ sơ và trạng thái người chơi
- **game\_logic:** Xử lý logic ghi nhận kết quả trận đấu
- **economy:** Quản lý token và cơ chế kinh tế trong game
- **nft\_items:** Phát hành và quản lý NFT vật phẩm
- **achievements:** Phát hành và quản lý NFT thành tích

### **4.4.2. Cấu trúc dữ liệu PlayerProfile**

Hồ sơ người chơi là đối tượng core của hệ thống:

```
struct PlayerProfile has key {  
    id: UID,  
    owner: address,  
    balance: u64,  
    streak: u64,  
    total_wins: u64,  
    total_losses: u64,  
    created_at: u64,  
}
```

Các trường dữ liệu:

- id: UID duy nhất của đối tượng
- owner: Địa chỉ ví sở hữu hồ sơ này
- balance: Số dư token trong game
- streak: Số ván thắng liên tiếp hiện tại
- total\_wins: Tổng số ván thắng từ trước đến nay
- total\_losses: Tổng số ván thua
- created\_at: Timestamp tạo hồ sơ

#### 4.4.3. Function ghi nhận phiên chơi

Hàm record\_session được gọi sau mỗi ván chơi để cập nhật trạng thái:

```
public fun record_session(  
    profile: &mut PlayerProfile,  
    win: bool,  
    ctx: &mut TxContext  
) {  
    if (win) {  
        profile.streak = profile.streak + 1;  
        profile.total_wins = profile.total_wins + 1;  
  
        // Thưởng token  
        profile.balance = profile.balance + REWARD_PER_WIN;  
  
        // Kiểm tra milestone thành tích  
        check_and_award_achievements(profile, ctx);  
    } else {  
        profile.streak = 0;  
        profile.total_losses = profile.total_losses + 1;  
    }  
}
```

Logic xử lý:

- Nếu thắng: tăng streak, total wins, cộng token thưởng, kiểm tra thành tích

- Nếu thua: reset streak về 0, tăng total losses
- Mọi thay đổi được ghi nhận bất biến trên blockchain

#### **4.4.4. Hệ thống NFT**

Hai loại NFT được triển khai với cấu trúc khác nhau:

##### **NFT vật phẩm (transferable):**

```
struct GameItem has key, store {
    id: UID,
    name: String,
    description: String,
    image_url: String,
    rarity: u8,
}
```

Có store ability nên có thể chuyển nhượng tự do.

##### **NFT thành tích (soulbound):**

```
struct Achievement has key {
    id: UID,
    milestone: u64,
    awarded_at: u64,
    owner: address,
}
```

Không có store ability nên không thể chuyển nhượng, gắn vĩnh viễn với người chơi.

### **4.5. Luồng xử lý dữ liệu đầu-cuối**

#### **4.5.1. Luồng bắt đầu phiên chơi**

1. Người chơi kết nối ví thông qua frontend
2. Frontend truy vấn PlayerProfile từ blockchain bằng địa chỉ ví
3. Nếu chưa có profile, hiển thị nút "Create Profile"
4. Người chơi tạo profile bằng cách ký giao dịch tạo mới
5. Blockchain tạo đối tượng PlayerProfile và gán ownership cho địa chỉ ví
6. Frontend tải lại profile và hiển thị thông tin người chơi
7. Người chơi có thể bắt đầu gameplay

#### **4.5.2. Luồng kết thúc ván chơi**

1. Gameplay kết thúc với kết quả thắng/thua xác định
2. Frontend tạo transaction gọi hàm record\_session
3. Gửi transaction đến ví để ký
4. Người chơi xác nhận transaction trong ví

5. Transaction được broadcast lên Sui Blockchain
6. Validator nodes xác thực và thực thi transaction
7. PlayerProfile được cập nhật với dữ liệu mới
8. Frontend nhận event hoàn thành và truy vấn lại profile
9. UI cập nhật hiển thị streak, balance mới

### 4.5.3. Luồng nhận thành tích

1. Sau khi thắng ván, smart contract kiểm tra milestone
2. Nếu đạt milestone mới (ví dụ: 10 ván thắng liên tiếp), tạo NFT Achievement
3. NFT được mint và transfer cho địa chỉ người chơi
4. Event được phát ra thông báo achievement mới
5. Frontend lắng nghe event và hiển thị popup chúc mừng
6. NFT xuất hiện trong kho achievement của người chơi

## 4.6. Xử lý bảo mật và chống gian lận

### 4.6.1. Các vector tấn công tiềm tàng

Do gameplay xử lý off-chain, hệ thống đối mặt với rủi ro:

- **Result manipulation:** Người chơi gửi kết quả thắng giả mạo
- **Replay attack:** Gửi lại giao dịch thắng cũ nhiều lần
- **Bot abuse:** Sử dụng bot để farm token tự động
- **Sybil attack:** Tạo nhiều tài khoản để lợi dụng faucet

### 4.6.2. Các biện pháp phòng chống

- **Ownership verification:** Smart contract kiểm tra gọi hàm phải đến từ chủ sở hữu profile
- **Rate limiting:** Giới hạn số lần claim faucet mỗi ngày
- **Server-side validation:** (Trong phiên bản tương lai) Xác thực kết quả qua server tin cậy
- **Cryptographic proof:** (Trong phiên bản tương lai) Yêu cầu proof of gameplay hợp lệ

Do giới hạn thời gian hackathon, phiên bản hiện tại chủ yếu dựa vào ownership verification cơ bản. Các cơ chế bảo mật nâng cao sẽ được triển khai trong các iteration tiếp theo.

# 5. TRIỂN KHAI VÀ KẾT QUẢ

## 5.1. Môi trường phát triển

Dự án được phát triển trong môi trường sau:

- **Blockchain:** Sui Testnet
- **Smart Contract:** Move compiler 1.x

- **Frontend:** Node.js 18+, React 18, Vite build tool
- **Ví:** Sui Wallet browser extension
- **SDK:** `@mysten/sui.js` version 0.4x+
- **IDE:** Visual Studio Code với Move Analyzer extension

## 5.2. Quy trình triển khai

### 5.2.1. Triển khai Smart Contract

Các bước triển khai smart contract lên Sui Testnet:

1. Viết và kiểm tra các module Move locally
2. Build package bằng lệnh `sui move build`
3. Chạy unit tests với `sui move test`
4. Publish lên testnet bằng `sui client publish`
5. Ghi lại Package ID và Object IDs của các shared objects
6. Cập nhật Package ID vào frontend configuration

### 5.2.2. Triển khai Frontend

Frontend được triển khai dưới dạng static web application:

1. Build production bundle với `npm run build`
2. Deploy lên hosting platform (Vercel, Netlify, hoặc IPFS)
3. Cấu hình environment variables cho Sui RPC endpoint
4. Kiểm tra kết nối với smart contract trên testnet
5. Thực hiện smoke testing các chức năng chính

## 5.3. Kết quả đạt được

### 5.3.1. Chức năng hoàn thành

Dự án đã triển khai thành công các chức năng sau:

- Gameplay bùng chun 2D hoạt động mượt mà với physics engine chính xác
- Tích hợp hoàn chỉnh với Sui Wallet cho xác thực và ký giao dịch
- Hệ thống hồ sơ người chơi on-chain với đầy đủ thông kê
- Cơ chế ghi nhận kết quả trận đấu tự động sau mỗi ván
- Hệ thống token trong game với faucet và reward mechanism
- NFT vật phẩm có thể mint và transfer
- NFT thành tích soulbound được tự động trao khi đạt milestone
- Giao diện người dùng responsive và trực quan

### 5.3.2. Chỉ số hiệu năng

Các chỉ số hiệu năng được đo đạc trên Sui Testnet:

Chỉ số	Giá trị
Frame rate gameplay	60 FPS
Độ trễ input	< 30ms
Thời gian tải trang	< 2s
Thời gian xác nhận giao dịch	1-2s
Gas cost mỗi giao dịch	~0.001 SUI
Thời gian query profile	< 500ms

Table 1: Bảng chỉ số hiệu năng hệ thống SuiChin

### 5.3.3. Thông kê sử dụng trong Hackathon

Trong thời gian demo tại hackathon:

- Số lượng profiles được tạo: 50+ tài khoản
- Tổng số ván chơi: 200+ sessions
- Số NFT thành tích được trao: 15 achievements
- Uptime hệ thống: 98%
- Không có transaction failed do lỗi smart contract

## 5.4. Đánh giá và bài học kinh nghiệm

### 5.4.1. Những điểm mạnh

- Kiến trúc hybrid giúp cân bằng tốt giữa hiệu năng và tính phi tập trung
- Sui Blockchain cho phép transaction finality rất nhanh, phù hợp cho game
- Move language với type system mạnh giúp tránh được nhiều lỗi bảo mật
- Ownership model của Sui rất phù hợp cho việc quản lý tài sản game

### 5.4.2. Những thách thức

- Học curve của Move language khá cao so với Solidity
- Documentation của Sui ecosystem còn đang được hoàn thiện
- Việc debug smart contract trên blockchain khó khăn hơn backend truyền thống
- Balance giữa decentralization và user experience vẫn là thách thức lớn

### 5.4.3. Bài học rút ra

- Cần phân tích kỹ phần nào nên on-chain, phần nào nên off-chain ngay từ đầu

- Unit testing cho smart contract là cực kỳ quan trọng do immutability
- User onboarding cho Web3 app cần được đầu tư công sức đáng kể
- Nên có fallback mechanism khi blockchain gặp sự cố tạm thời

## 6. HẠN CHẾ VÀ HƯỚNG PHÁT TRIỂN TƯƠNG LAI

### 6.1. Hạn chế của phiên bản hiện tại

#### 6.1.1. Hạn chế về chức năng

- Chưa có chế độ multiplayer thời gian thực, chỉ chơi với bot AI
- Chưa có bảng xếp hạng toàn cầu để người chơi so sánh thành tích
- Hệ thống kinh tế chưa có tokenomics hoàn chỉnh với cơ chế đốt token
- Chưa có marketplace tích hợp để giao dịch NFT vật phẩm
- Gameplay còn đơn giản, chưa có nhiều mode và chiến thuật phức tạp

#### 6.1.2. Hạn chế về bảo mật

- Chưa có cơ chế xác thực kết quả gameplay phía server
- Dễ bị lợi dụng bởi bot tự động nếu không có captcha
- Chưa có rate limiting đầy đủ cho các actions
- Chưa audit smart contract bởi bên thứ ba

#### 6.1.3. Hạn chế về môi trường

- Chỉ triển khai trên testnet, chưa sẵn sàng cho mainnet production
- Chưa tối ưu gas cost cho các transaction phức tạp
- Chưa có monitoring và alerting system cho smart contract

### 6.2. Tầm nhìn dài hạn

#### 6.2.1. Mở rộng gameplay

- Thêm các loại chún với thuộc tính đặc biệt
- Nhiều map với địa hình và vật cản khác nhau
- Power-ups và abilities đặc biệt
- Campaign mode với storyline
- Boss battles và PvE content

#### 6.2.2. Cross-chain integration

- Bridge NFT sang các chain khác (Ethereum, Polygon)
- Multi-chain marketplace
- Cross-chain tournaments

### 6.2.3. Metaverse elements

- Virtual world nơi người chơi gặp gỡ và giao lưu
- Customizable avatars
- Virtual land ownership
- Integration với các metaverse platforms khác

### 6.2.4. DAO governance

- Phát hành governance token
- Cho phép community vote về game mechanics
- Treasury management bởi DAO
- Decentralized game development

## 6.3. Khả năng áp dụng cho các dự án khác

Kiến trúc và các giải pháp kỹ thuật của SuiChin có thể được áp dụng cho:

- Các game casual khác muốn tích hợp blockchain
- Hệ thống loyalty program với NFT rewards
- Ứng dụng giáo dục gamification
- Social platform với on-chain reputation
- Prediction markets và betting platforms

# 7. KẾT LUẬN

## 7.1. Tổng kết đóng góp

Dự án SuiChin đã thành công trong việc chứng minh tính khả thi của việc xây dựng game Web3 trên nền tảng Sui Blockchain. Các đóng góp chính bao gồm:

1. **Mô hình kiến trúc hybrid:** Đề xuất và triển khai kiến trúc cân bằng hiệu quả giữa off-chain gameplay và on-chain asset management, đạt được cả trải nghiệm người dùng mượt mà lẫn tính minh bạch của blockchain
2. **Triển khai thực tế trên Sui:** Cung cấp case study cụ thể về cách sử dụng Move language, Sui object model và Sui SDK để xây dựng ứng dụng game hoàn chỉnh
3. **Giải pháp NFT phân tầng:** Thiết kế và triển khai hai loại NFT với mục đích khác nhau (transferable items vs soulbound achievements), phù hợp với nhu cầu game economy
4. **Open source reference:** Tạo ra một dự án mã nguồn mở có thể làm tài liệu tham khảo cho các developers khác muốn xây dựng trên Sui

## 7.2. Ý nghĩa trong bối cảnh Web3 Gaming

SuiChin đại diện cho một hướng tiếp cận thực tế trong việc xây dựng Web3 games, không cố gắng đưa mọi thứ lên blockchain mà tập trung vào việc tận dụng blockchain cho những gì nó làm tốt nhất: ownership, transparency và trustlessness. Cách tiếp cận này có thể giúp ngành Web3 gaming tránh được các cạm bẫy của over-engineering và tập trung vào trải nghiệm người dùng thực sự.

## 7.3. Lời cảm ơn

Chúng tôi xin gửi lời cảm ơn chân thành đến:

- Ban tổ chức Sui Campus Hackathon đã tạo cơ hội cho chúng tôi tìm hiểu và phát triển trên Sui Blockchain
- Mysten Labs và Sui Foundation đã cung cấp documentation, tools và support community tuyệt vời
- Các mentors đã góp ý và hướng dẫn trong suốt quá trình phát triển
- Học viện Công nghệ Bưu chính Viễn thông (PTIT) đã tạo điều kiện cho đội nghiên cứu
- Cộng đồng Sui developers đã chia sẻ kinh nghiệm và best practices

## 7.4. Lời kết

Dự án SuiChin không chỉ là một game, mà là một proof-of-concept cho thấy Web3 gaming có thể đạt được sự cân bằng giữa decentralization và usability. Chúng tôi hy vọng dự án này sẽ truyền cảm hứng cho nhiều developers khác khám phá tiềm năng của Sui Blockchain và đóng góp vào sự phát triển của hệ sinh thái Web3 gaming.

Với nền tảng vững chắc đã được xây dựng, SuiChin sẵn sàng cho các giai đoạn phát triển tiếp theo hướng tới việc trở thành một game Web3 có quy mô và cộng đồng người chơi thực sự. Chúng tôi tin rằng tương lai của gaming nằm ở sự kết hợp hài hòa giữa entertainment value và true digital ownership.

## TÀI LIỆU THAM KHẢO

- [1] Mysten Labs. (2024). *Sui Documentation*. <https://docs.sui.io/>
- [2] Mysten Labs. (2024). *Move Language Book*. <https://letsmovesui.com/>
- [3] Sui Foundation. (2024). *Sui Campus Hackathon Training Materials*. Sui Campus Hackathon Program.
- [4] Blackshear, S., et al. (2022). *Move: A Language With Programmable Resources*. Mysten Labs Technical Papers.
- [5] Wood, G. (2014). *Ethereum: A Secure Decentralised Generalised Transaction Ledger*. Ethereum Project Yellow Paper.
- [6] Buterin, V. (2021). *The Limits to Blockchain Scalability*. Vitalik.ca Blog. <https://vitalik.ca/general/2021/05/23/scaling.html>
- [7] Sui Foundation. (2023). *Sui Blockchain Whitepaper*. <https://sui.io/whitepaper>

- [8] Dfinity Foundation. (2020). *Internet Computer Protocol Architecture*. IC Wiki.
- [9] Yaish, A., et al. (2023). *Uncle Maker: (Time)Stamping Out The Competition in Ethereum*. Financial Cryptography and Data Security Conference.
- [10] Chen, J., & Micali, S. (2019). *Algorand: A Secure and Efficient Distributed Ledger*. Theoretical Computer Science, 777, 155-183.