**TRƯỜNG ĐẠI HỌC THỦY LỢI**

**KHOA CÔNG NGHỆ THÔNG TIN**

****

**BÁO CÁO BÀI TẬP LỚN MÔN HỌC**

**CÔNG NGHỆ PHẦN MỀM**

***Đề tài:***

**Nhóm sinh viên thực hiện: Nhóm …**

**Thành viên nhóm:** …

**Giảng viên hướng dẫn*:***

***Hà Nội, tháng 10 năm 2023***

NHIỆM VỤ 3/4: SPRINT 1/2

# **I. Tài liệu đặc tả yêu cầu phần mềm**

## 1.1 Biểu đồ Phân rã chức năng (FDD)



## 1.2 Biểu đồ Luồng dữ liệu (DFD)

### 1.2.1 DFD Start Game



### 1.2.2 DFD Game Over

**

### 1.2.3 DFD Level Up



### 1.2.4 DFD Trang bị vũ khí



### 1.2.5 DFD Quản lý điểm

### 1.2.6 DFD Bật/tắt âm thanh



### 1.2.7 DFD Sửa bài đăng

…

### 1.2.8 DFD Xóa bài đăng

…

### 1.2.9 DFD Xem bài đăng

…

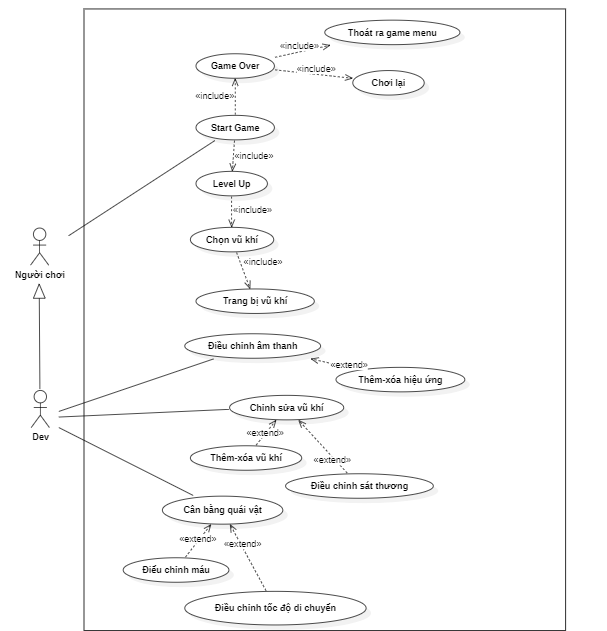
### 1.2.10 DFD Bình luận

…

### 1.2.11 DFD Xóa bình luận

…

## 1.3 Sơ đồ Use case tổng quát



## 1.4 Miêu tả chi tiết các Use cases

### 1.4.1 Use case Đăng ký

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID | 01 | | |
| Use Case Name | Start Game | | |
| Created By: | … | Last Updated By | … |
| Date Created: | … | Date Last Updated | … |

|  |  |
| --- | --- |
| Actor | Người chơi |
| *Description* | Chức năng cho phép người chơi chọn chức năng từ game menu |
| *Trigger* | Chức năng này được sử dụng khi người chơi khởi động game |
| *Pre-conditions* | Người chơi tải game |
| *Post conditions* | Người chơi khởi động game và chọn “Start” |
| *Flow of Events* | 1. Người chơi tải game  2. Người chơi khởi động game  3. Hệ thống hiển thị giao diện game menu  4. Người chơi chọn “Start” để bắt đầu chơi  5. Hệ thống bắt đầu trò chơi |

### 1.4.2 Use case Game Over

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID | 02 | | |
| Use Case Name | Game Over | | |
| Created By: | … | Last Updated By | … |
| Date Created: | … | Date Last Updated | … |

|  |  |
| --- | --- |
| Actor | Người chơi |
| *Description* | Chức năng thông báo điểm, trò chơi đã kết thúc nếu nhân vật người chơi điều khiển hết điểm sinh mệnh. |
| *Trigger* | Chức năng này được sử dụng khi nhân vật người chơi điều khiển hết sinh mệnh. |
| *Pre-conditions* | Người chơi chọn “Start” và bắt đầu chơi game |
| *Post conditions* | Người chơi khởi động game và chọn “Start” |
| *Flow of Events* | 1. Người chơi chọn “Start” và bắt đầu chơi  2. Hệ thống hiển thị thanh sinh mệnh  3. Nếu nhân vật hết sinh lực hệ thống hiển thị điểm, thông báo Game Over và cho người chơi lựa chọn chơi lại hoặc thoát ra game menu |

### 1.4.3 Use case Level Up

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID | 03 | | |
| Use Case Name | Level Up | | |
| Created By: | … | Last Updated By | … |
| Date Created: | … | Date Last Updated | … |

|  |  |
| --- | --- |
| Actor | Hệ thống |
| *Description* | Chức năng lên cấp nhân vật |
| *Trigger* | Chức năng này được sử dụng khi nhân vật đủ điểm kinh nghiệm từ việc tiêu dệt quai vật |
| *Pre-conditions* | Nhân vật tích lũy đủ điểm kinh nghiệm |
| *Post conditions* |  |
| *Flow of Events* | 1. Người chơi điều khiển nhân vật tiêu diệt quái vật  2. Hệ thống tăng điểm kinh nghiệm theo số quai vật bị tiêu diệt  3. Khi kinh nghiệm đã đủ, hệ thống hiện lên bảng chọn vũ khí và kinh nghiệm reset. |

### 1.4.4 Use case Trang bị vũ khí

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID | 04 | | |
| Use Case Name | Trang bị vũ khí | | |
| Created By: | … | Last Updated By | … |
| Date Created: | … | Date Last Updated | … |

|  |  |
| --- | --- |
| Actor | Hệ thống |
| *Description* | Chức năng thể hiện vũ khí cạnh nhân vật sau khi người chơi chọn vũ khí |
| *Trigger* | Chức năng này được sử dụng khi người chơi đã chọn vũ khí sau khi lên cấp |
| *Pre-conditions* | Người chơi đã chọn vũ khí |
| *Post conditions* |  |
| *Flow of Events* | 1. Người chơi sau khi lên cấp sẽ được chọn vũ khí  2. Sau khi người chơi đã chọn xong vũ khí, hệ thống sẽ hiển thị vũ khí đã chọn cạnh nhân vật |

### 1.4.5 Use case Bình luận

…

### 1.4.6 Use case Xóa bình luận

…

### 1.4.7 Use case Xem bài đăng

…

### 1.4.8 Use case Tạo bài đăng

…

### 1.4.9 Use case Sửa bài đăng

…

### 1.4.10 Use case Xóa bài đăng

…

## 1.5 Sơ đồ Trình tự (SD)

### 1.5.1 SD Đăng Nhập

…

1.5.2 SD Đăng ký

…

### 1.5.3 SD Đổi mật khẩu



1.5.4 SD Tạo bài đăng

…

1.5.5 SD Xóa bài đăng

…

### 1.5.6 SD Sửa bài đăng

…

### 1.5.7 SD Xem bài đăng

…

### 1.5.8 SD Bình luận

…

### 1.5.9 SD Xóa bình luận

…

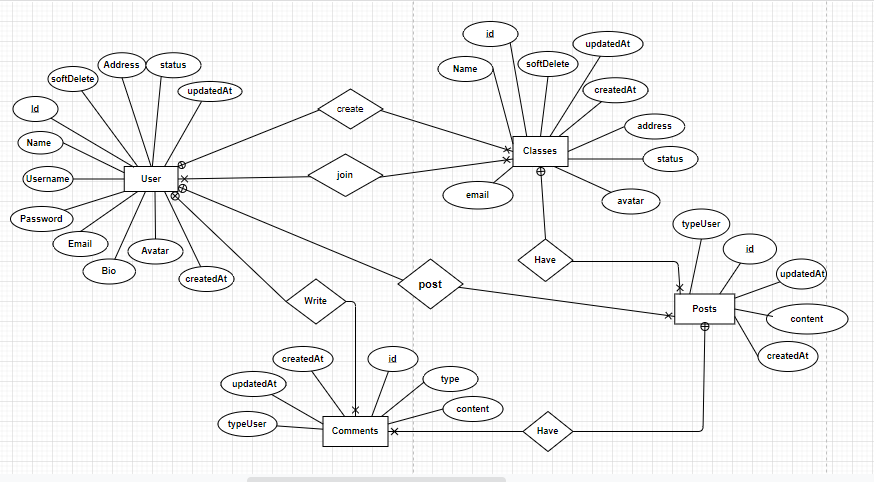
### 1.5.10 SD Tạo lớp tham gia

…

### 1.5.11 SD Tham gia lớp

…

## 1.6 Biểu đồ quan hệ thực thể (ERD)



# **II. Thiết kế phần mềm**

## 1. **Thiết kế Kiến trúc**

<< HÌNH VẼ MINH HOẠ VÀ GIẢI THÍCH CÁC THÀNH PHẦN TRONG KIẾN TRÚC CỦA PHẦN MỀM >>

## 2. **Thiết kế Hướng đối tượng**

<< NẾU PHẦN MỀM ĐƯỢC PHÁT TRIỂN THEO HƯỚNG ĐỐI TƯỢNG THÌ VẼ HÌNH MINH HOẠ VÀ GIẢI THÍCH CÁC LỚP, CŨNG NHƯ MỐI LIÊN HỆ GIỮA CÁC LỚP >>

VÍ DỤ



## 3. **Thiết kế Dữ liệu**

<<DỰA VÀO BIỂU ĐỒ QUAN HỆ THỰC THỂ Ở TRÊN>>

### 3.1 Các thực thể và thuộc tính

* **users**(id, username, password, email, bio, avatar, status, type, age, sex, birthday, created\_at, updated\_at, soft\_delete)
* **posts**(id, status, content, created\_at, updated\_at, soft\_delete)
* **classes**(id, name, cover, status, referral\_code, created\_at, updated\_at, soft\_delete)
* **comments**(id, type\_comment, content, created\_at, updated\_at, soft\_delete)

### 3.2 Mối liên kết

* **users** và **posts** là liên kết 1 - n
* **users** và **classes** là liên kết m - n
* **users** và **comments** là liên kết 1 - n
* **posts** và **comments** là liên kết 1 - n
* **classes** và **posts** là liên kết 1 - n

### 3.3 Chuyển sang quan hệ

* Liên kết 1 - N → Khóa chính của quan hệ bên một trở thành khóa ngoại của quan hệ bên nhiều.
* Liên kết M - N → Tạo bảng quan hệ mới, chuyển khóa chính của hai quan hệ phía M và N thành khóa ngoại của quan hệ mới. Khóa chính của quan hệ mới là sự kết hợp của hai khóa ngoại.

Kết quả:

* **users**(id, username, password, email, bio, avatar, status, type, age, sex, birthday, created\_at, update\_at, soft\_delete)
* **posts**(id, status, content, created\_at, updated\_at, soft\_delete, owner\_id, class\_id)
* **classes**(id, name, cover, status, referral\_code, created\_at, updated\_at, soft\_delete)
* **comments**(id, type\_comment, content, created\_at, updated\_at, soft\_delete, owner\_id, topic\_id)
* **student\_classes**(id, created\_at, updated\_at, soft\_delete, status, user\_id, class\_id)

### 3.4 Chuẩn hóa dữ liệu

<<CÓ THỂ BỎ QUA NẾU CHƯA HỌC Ở CÁC MÔN HỌC TRƯỚC>>

#### 3.4.1 Xác định phụ thuộc hàm

* id xác định một người dùng duy nhất bao gồm tên người dùng, mật khẩu, email, lý lịch, ảnh đại diện, trạng thái, địa chỉ, ngày tạo, ngày cập nhật, xóa mềm.
* id xác định một bài đăng duy nhất bao gồm các thông tin về bài đăng : nội dung, kiểu người dùng, thời gian đăng bài, thời gian chỉnh sửa, xóa mềm.
* id xác định một lớp học duy nhất gồm thông tin lớp học: tên lớp, email, hình đại diện, trạng thái lớp học, địa chỉ, thời gian tạo, thời gian chỉnh sửa, xóa mềm.
* id xác định một bình luận hoa duy nhất gồm các thông tin liên quan đến bình luận đó: loại bình luận, nội dung, loại người dùng, thời gian bình luận, thời gian chỉnh sửa, xóa mềm, mã người dùng và mã bài đăng.

#### 3.4.2 Xét các dạng chuẩn

* **users**(id, username, password, email, bio, avatar, status, type, age, sex, birthday, created\_at, updated\_at, soft\_delete)

Đặt id = A, username = B, password = C, email = D,bio = E, avatar = F, status = G, type = H, age = I, created\_at = K, updated)at = N, soft\_delete = M, sex = O, birthday = Q

Phụ thuộc hàm:

F = {A → B, A → C, A → D, A → E, A → F, A → G, A → H, A → I, A → K, A → N, A → M, A → O, A → Q}

=> Đạt 1NF vì không có thuộc tính đa trị

=> Đạt 2NF vì các thuộc tính không khóa phụ thuộc đầy đủ vào khóa chính A

=> Đạt 3NF vì các thuộc tính không khóa phụ thuộc trực tiếp vào khóa chính A

=> Đạt BCNF vì không có thuộc tính khóa nào phụ thuộc vào thuộc tính không khóa

* **posts**(id, status, content, created\_at, updated\_at, soft\_delete, owner\_id, class\_id)

…

* **classes**(id, name, cover, status, referral\_code, created\_at, updated\_at, soft\_delete)

…

* **comments**(id, type\_comment, content, created\_at, updated\_at, soft\_delete, owner\_id, topic\_id )

…

* student\_classes(id, created\_at, updated\_at, soft\_delete, status, user\_id, class\_id )

…

### 3.5 Sơ đồ quan hệ

<<HÌNH VẼ SƠ ĐỒ QUAN HỆ >>

## 4. **Thiết kế Giao diện Người dùng**

HÌNH VẼ GIAO DIỆN CỦA PHẦN MỀM. CÓ THỂ SỬ DỤNG MỘT SỐ NỀN TẢNG CHO PHÉP THIẾT KẾ GIAO DIỆN NGƯỜI DÙNG NHƯ: [NinjaMock](https://ninjamock.com/), [Figma](https://www.figma.com/), [MockingBot](https://mockingbot.in/), [MarvelApp](https://marvelapp.com/examples), [Sketch](https://www.sketch.com/), Adobe Photoshop, …

**NGHIÊM CẤM CHỤP MÀN HÌNH CHƯƠNG TRÌNH CÓ SẴN**

# **III. Mã nguồn**

<< LINK MÃ NGUỒN TRÊN GITHUB, NẾU CÓ>>

# **IV. Tài liệu kiểm thử**

* Tham khảo báo cáo [kiểm thử giao diện](https://docs.google.com/spreadsheets/d/1FCA8NTF_SsqQ2uHMDUXzuczcVpzEwfFQPNqQ6j1Ih9Y/edit?usp=sharing)
* Tham khảo báo cáo [kiểm thử chức năng](https://docs.google.com/spreadsheets/d/1JM-fTYkRxvUmpwPzZ_QcdknETALVys91ZCZwdBbAj6U/edit?usp=sharing)