
Báo cáo #2: Thiết kế hệ thống

ClassPin

Nhập môn Công nghệ phần mềm
(CSC13002)

Tên nhóm: SugarDEV

1. Vũ Lê Tuấn
2. Lê Quốc Đạt
3. Phạm Hoàng An
4. Trương Công Quốc Cường

Thành phố Hồ Chí Minh, 17/11/2020

Lịch sử cập nhật

| STT | Ngày | Phiên bản | Mô tả | Tác giả |
|-----|------|-----------|-------|---------|
| 1 | | | | |
| 2 | | | | |
| 3 | | | | |
| ... | | | | |

Phân tích đóng góp cá nhân

| STT | Thành viên | Công việc | Đóng góp |
|-----|------------------------|---|----------|
| 1 | Vũ Lê Tuấn | Giao thức mạng, yêu cầu phần cứng, biểu đồ lớp, quản trị dự án và kế hoạch làm việc | 25% |
| 2 | Lê Quốc Đạt | Kiến trúc hệ thống, cấu trúc dữ liệu, quản trị dự án và kế hoạch làm việc | 25% |
| 3 | Phạm Hoàng An | Nhận diện hệ thống con, luồng điều khiển, thuật toán | 25% |
| 4 | Trương Công Quốc Cường | Ánh xạ các phần của hệ thống với phần cứng, lưu trữ dữ liệu lâu dài | 25% |

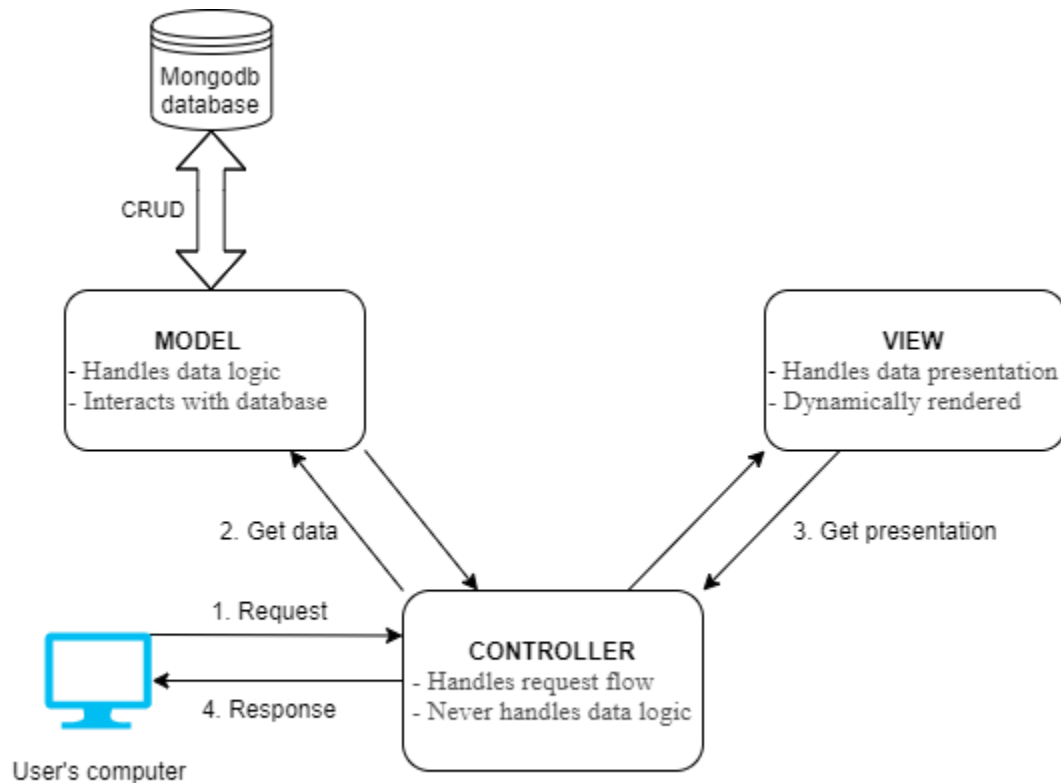
Mục lục

Báo cáo thiết kế hệ thống

| | |
|--|----|
| 1. Thiết kế kiến trúc và hệ thống..... | 6 |
| 1.1. Kiến trúc hệ thống..... | 6 |
| 1.2. Nhận diện hệ thống con | 8 |
| 1.3. Ánh xạ các phần của hệ thống với phần cứng | 9 |
| 1.4. Lưu trữ dữ liệu lâu dài | 10 |
| 1.5. Giao thức mạng..... | 12 |
| 1.6. Luồng điều khiển (Global Control Flow) | 13 |
| 1.7. Yêu cầu phần cứng..... | 15 |
| 2. Biểu đồ lớp..... | 16 |
| 2.1. Biểu đồ lớp..... | 16 |
| 2.2. Đặc tả các lớp..... | 16 |
| 2.2.1. Lớp C1 | 16 |
| 2.2.2. Lớp C2 | 17 |
| 3. Thuật toán và cấu trúc dữ liệu..... | 19 |
| 3.1. Các thuật toán | 19 |
| 3.2. Các cấu trúc dữ liệu | 19 |
| 4. Quản trị dự án và kế hoạch làm việc..... | 20 |
| 4.1. Tổng hợp đóng góp từ các thành viên của nhóm..... | 20 |
| 4.2. Báo cáo tiến độ và điều phối dự án..... | 20 |
| 4.3. Kế hoạch thực hiện | 21 |
| 4.4. Phân rã trách nhiệm (Breakdown of Responsibilities) | 21 |
| 5. Tham khảo | 22 |

1. Thiết kế kiến trúc và hệ thống

1.1. Kiến trúc hệ thống



Model - View - Controller

Hình 1.1. Kiến trúc hệ thống MVC

*** Hệ thống sử dụng mô hình MVC, chia thành 3 phần:**

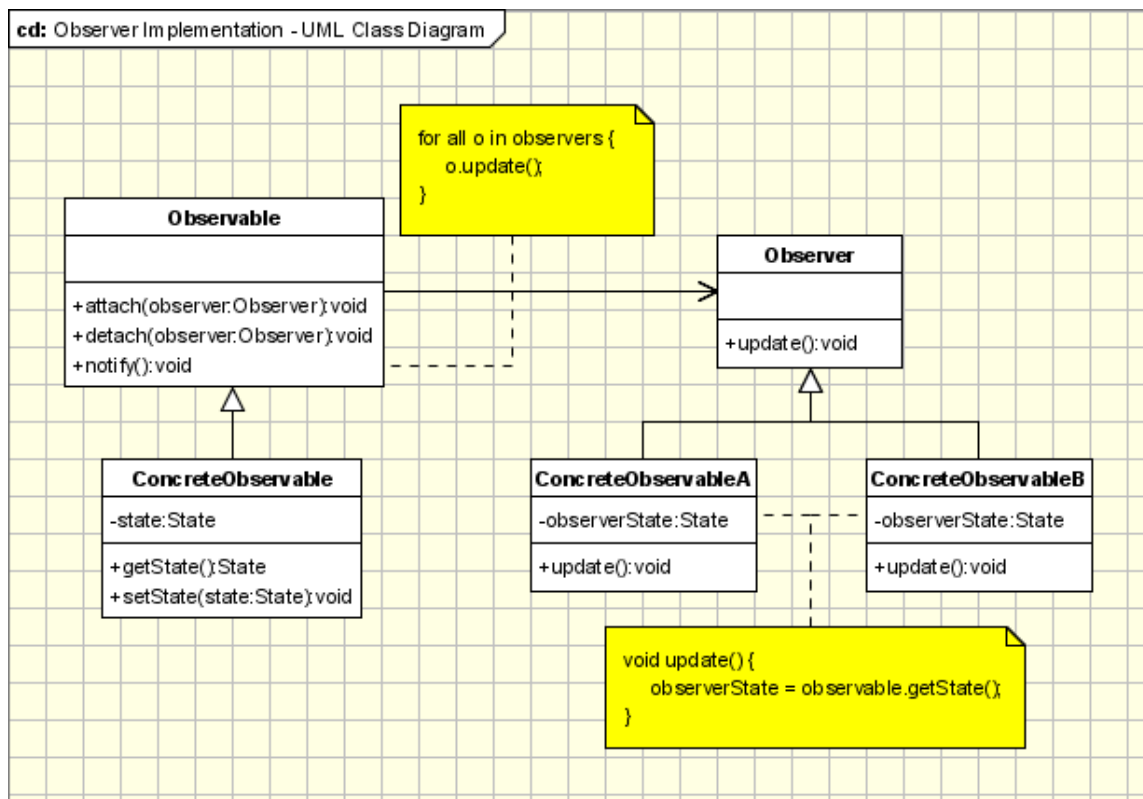
- Model: Chịu trách nhiệm xử lý data, model sẽ tương tác với database như create, retrieve, update, delete. Model không quan tâm đến việc xử lý request của client (các bước xử lý khi thành công, thất bại).

- Controller: Chịu trách nhiệm xử lý toàn bộ request từ client và cho biết request sẽ yêu cầu những gì. Controller sẽ tương tác với model dựa trên thông tin của request đó. Controller sẽ

không tương tác trực tiếp với data logic hay nói cách khác nó không quan tâm đến việc xử lý dữ liệu như thế nào, nó sẽ nói cho model biết việc cần làm là gì.

- View: Có nhiệm vụ biểu diễn dữ liệu mà controller gửi đến, view sẽ như là một file template và render HTML dựa trên dữ liệu mà controller gửi đến. View không quan tâm đến việc xử lý dữ liệu mà chỉ quan tâm đến việc biểu diễn dữ liệu như thế nào. View sẽ gửi biểu diễn dữ liệu lại cho controller và controller sẽ gửi phản hồi lại cho client.

* Mẫu thiết kế: Observer Pattern



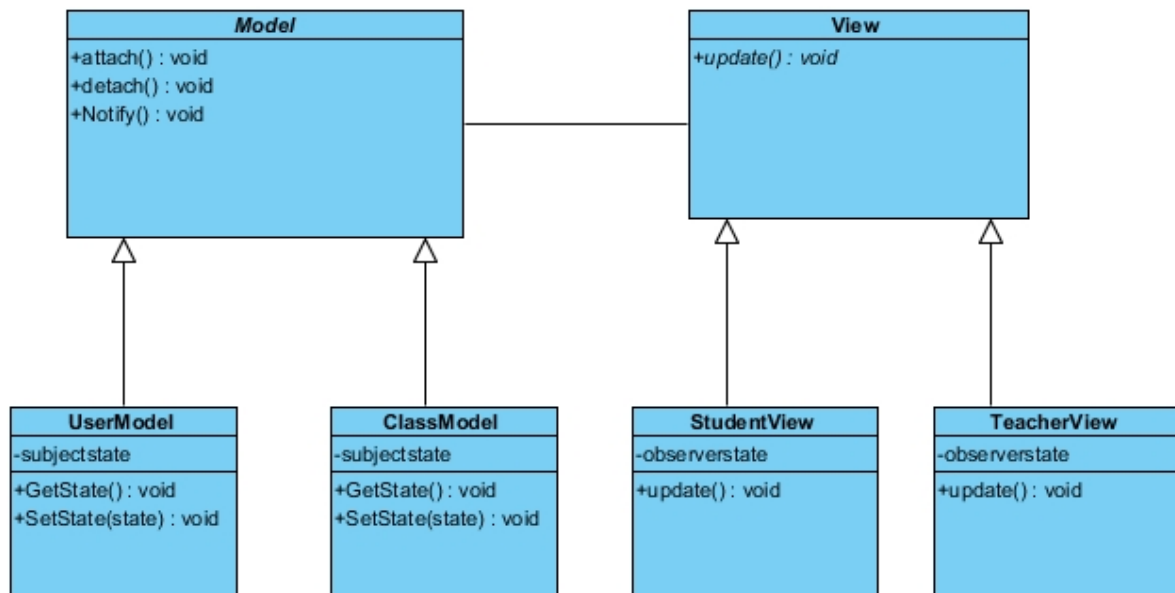
Hình 1.2. Observer Design Pattern

- Mẫu thiết kế Observer là một trong hai mươi ba mẫu thiết kế nổi tiếng mô tả cách giải quyết các vấn đề thiết kế lặp lại để thiết kế phần mềm hướng đối tượng linh hoạt và có thể tái sử dụng, tức là các đối tượng dễ thực hiện hơn, dễ thay đổi hơn, kiểm tra và sử dụng lại.

- Observer là một mẫu thiết kế giúp ta giải quyết các vấn đề sau:

- + Tạo mối liên hệ một – nhiều giữa Subject và các Observer.
- + Đảm bảo khi một đối tượng Subject được thay đổi, các đối tượng Observer sẽ được tự động cập nhật.
- + Một đối tượng có thể mở thông báo cho một số đối tượng khác.

*** Áp dụng Observer design pattern cho hệ thống:**



Hình 1.3. Áp dụng Observer vào hệ thống.

*** Mô tả:**

- Model sẽ đóng vai trò là một Subject và View sẽ giống như một Observer. Model sẽ giữ các dữ liệu và trạng thái dữ liệu, View sẽ đặc trưng cho những màn hình khác nhau, View sẽ lấy data từ Model khi dữ liệu được cập nhật.
- Mô hình tạo ra mối quan hệ một – nhiều giữa Subject và các Observer, vì vậy bất cứ lúc nào ta thay đổi dữ liệu của UserModel, ClassModel thì trạng thái của Model sẽ thay đổi thì tất cả các StudentView và TeacherView sẽ được thông báo và cập nhật ngay lập tức.

1.2. Nhận diện hệ thống con

Biểu đồ Component (UML)

Web interface là giao diện người dùng, gồm các file html để vẽ giao diện.

Các service trên web là các chức năng để người dùng yêu cầu thực hiện, khi 1 chức năng được gửi đi từ phía người dùng sẽ đến Controller trên máy chủ để quản lý các yêu cầu này, sau đó gọi các Service trên máy chủ và sử dụng Nodejs để hiển thị chức năng trên giao diện, các service có thể truy cập dữ liệu gián tiếp qua Database Handler.

1.4. Lưu trữ dữ liệu lâu dài

Sử dụng MongoDB để lưu trữ cơ sở dữ liệu.

Cấu trúc dữ liệu trong file .json:

```
1 {
2   "TestSubmission": [
3     {"idTestSubmission": "", "idStudent": "", "idTest": "", "listIDQuestion": [{"idques": "idchoice"}], "submitDate": ""}
4   ],
5   "users": [
6     {"id": "", "password": "hash(21321)", "email": "rqwrq", "dateOfBirth": "", "Name": ""}
7   ],
8   "classroom": [
9     {"id": "", "className": "", "listidUser": [{}], "startDate": "", "idTeacher": ""}
10  ],
11  "Test": [
12    {"id": "", "idclassroom": "", "idquestion": [{}], "description": "", "startTime": "", "endTime": "", "duration": ""},
13  ],
14  "question": [
15    {"id": "", "content": "", "listchoice": [{"idluachon": ""}], "idAnswer": "", "point": ""},
16  ],
17  "choice": [
18    {"id": "", "description": ""}
19  ],
20  "ClassTopic": [
21    {"id": "", "idUser": "", "topicContent": "", "idTest": ""}
22  ]
23 }
```

Mô tả lược đồ:

| Tên lược đồ | Thuộc tính | Kiểu dữ liệu |
|----------------|------------------|--------------------------|
| TestSubmission | idTestSubmission | Chuỗi ký tự |
| | idStudent | Chuỗi ký tự |
| | idTest | Chuỗi ký tự |
| | listIDQuestion | Mảng idquestion - choice |
| | submitDate | Chuỗi ký tự Date |
| users | id | Chuỗi ký tự |
| | password | Chuỗi ký tự (đã hash) |
| | email | Chuỗi ký tự |

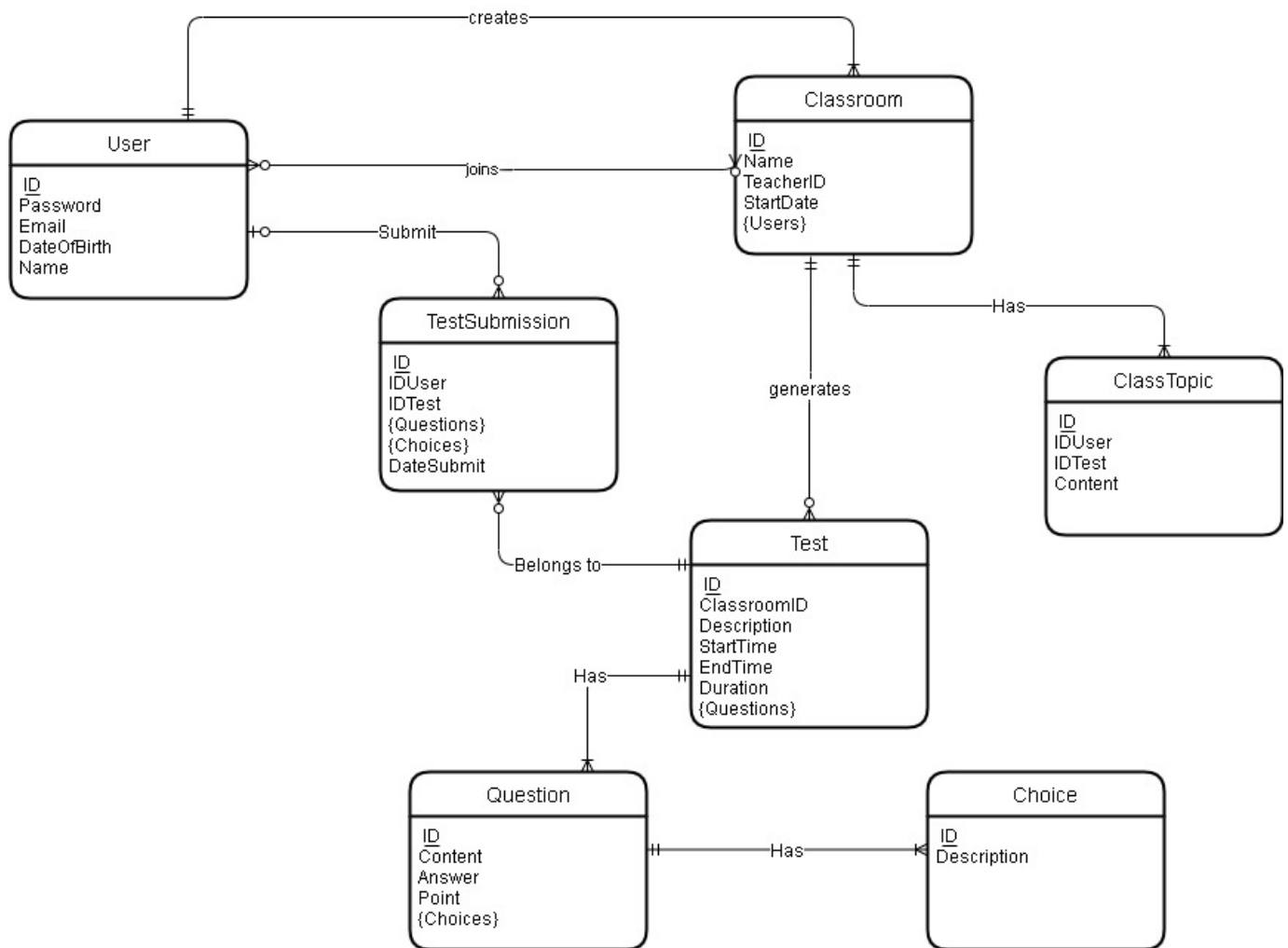
| | | |
|------------|-------------|----------------------|
| | DateOfBirth | Chuỗi ký tự Date |
| | Name | Chuỗi ký tự |
| classroom | id | Chuỗi ký tự |
| | className | Chuỗi ký tự |
| | listidUser | Mảng id của users |
| | startDate | Chuỗi ký tự Date |
| | idTeacher | Id của users |
| Test | id | Chuỗi ký tự |
| | idclassroom | Id của classroom |
| | idquestion | Mảng id của question |
| | description | Chuỗi ký tự |
| | startTime | Chuỗi ký tự Datetime |
| | endTime | Chuỗi ký tự Datetime |
| question | duration | Số thực |
| | id | Chuỗi ký tự |
| | content | Chuỗi ký tự |
| | listchoice | Mảng id của choice |
| | Answer | Chuỗi ký tự |
| choice | point | Số thực |
| | id | Chuỗi ký tự |
| | description | Chuỗi ký tự |
| classTopic | id | Chuỗi ký tự |
| | idUsers | Id của users |
| | Content | Chuỗi ký tự |
| | idTest | Id của test |

Bảng khóa:

| Tên lược đồ | Khóa chính | Khóa ngoại | |
|----------------|------------------|----------------|-----------|
| | | Tên khóa ngoại | Liên kết |
| TestSubmission | idTestSubmission | idStudent | users(id) |
| | | listIDquestion | Test(id) |
| users | id | | |

| | | | |
|------------|----|-------------|---------------|
| classroom | id | listIDUser | Users(id) |
| Test | id | idclassroom | Classroom(id) |
| question | id | listchoice | Choice(id) |
| choice | id | | |
| ClassTopic | id | idUser | Users(id) |
| | | idTest | Test(id) |

Conceptual Model:



1.5. Giao thức mạng

Sử dụng giao thức HTTP cho tầng ứng dụng và giao thức TCP cho tầng transport.

Vì HTTP là giao thức truyền tải siêu văn bản được sử dụng trong www dùng để truyền tải dữ liệu giữa Web server đến các trình duyệt Web và ngược lại, sử dụng chủ yếu port 80. ClassPin là một website, hoạt động trên trình duyệt web nên sử dụng giao thức HTTP.

Giao thức HTTP hoạt động dựa trên mô hình Client-Server, các máy tính của người dùng là Client và gửi yêu cầu lên Server, Server sẽ đọc yêu cầu và phản hồi cho các Client.

Nhóm không dùng HTTPS vì không có chứng chỉ SSH, điều này làm giảm khả năng bảo mật của trang web.

Đối với tầng transport, sử dụng giao thức TCP để đảm bảo tính chính xác và hoàn thiện của dữ liệu. Đối với giao thức TCP các máy tính phải tạo kết nối với nhau (bắt tay) để có thể truyền dữ liệu cho nhau, TCP có cơ chế kiểm soát tắc nghẽn và kiểm soát luồng để chắc chắn dữ liệu truyền đi không làm quá tải bên nhận. Các gói tin gửi thông qua TCP được đảm bảo chính xác và đúng thứ tự.

1.6. Luồng điều khiển (Global Control Flow)

- Thứ tự thực hiện: Hệ thống của nhóm là hệ thống hướng theo thủ tục và thực thi theo kiểu “tuyến tính”, trong đó người dùng mỗi lần đều phải trải qua các bước giống nhau.
- Phụ thuộc thời gian: Hệ thống có sử dụng bộ hẹn giờ khi user học sinh làm bài kiểm tra.
- Hệ thống của nhóm là hệ thống thời gian thực không có chu kỳ.

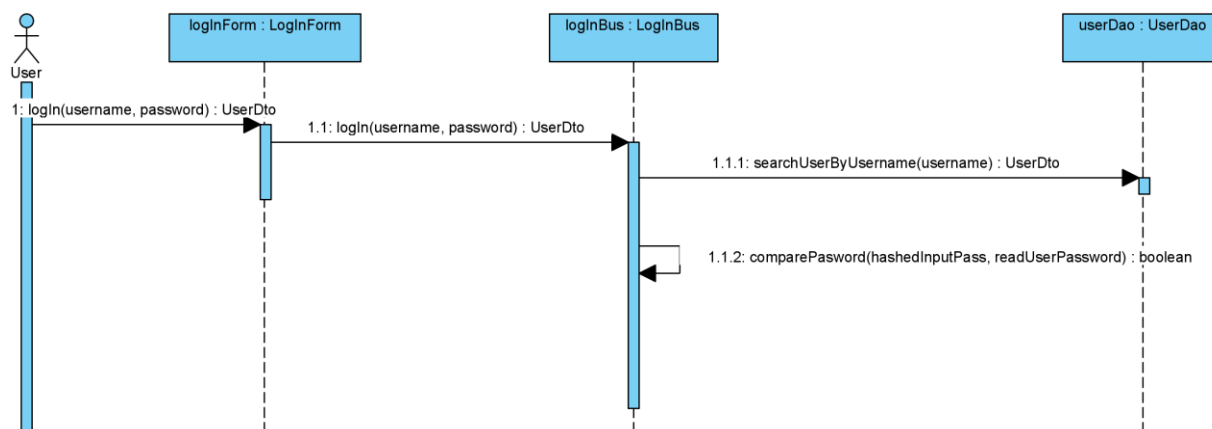


Figure 1. Đăng nhập

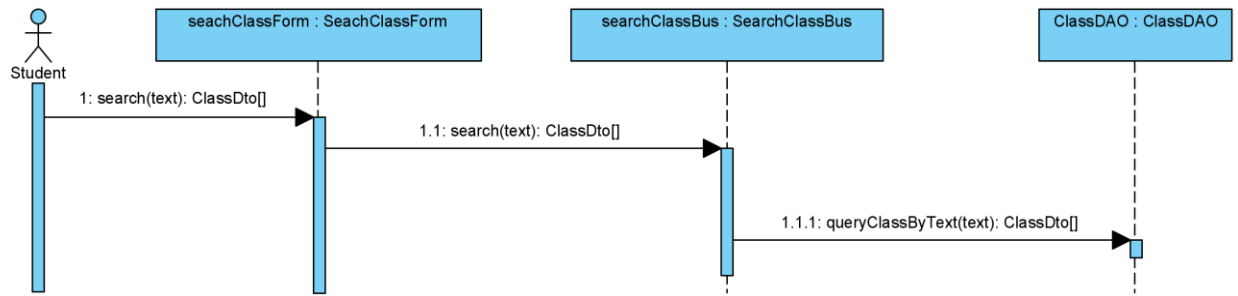


Figure 2. Tìm Lớp

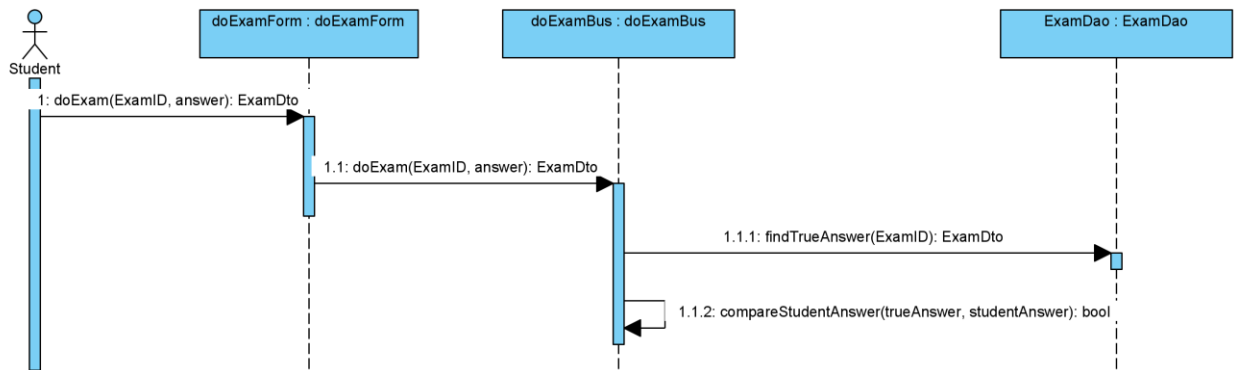


Figure 3. Học sinh làm bài

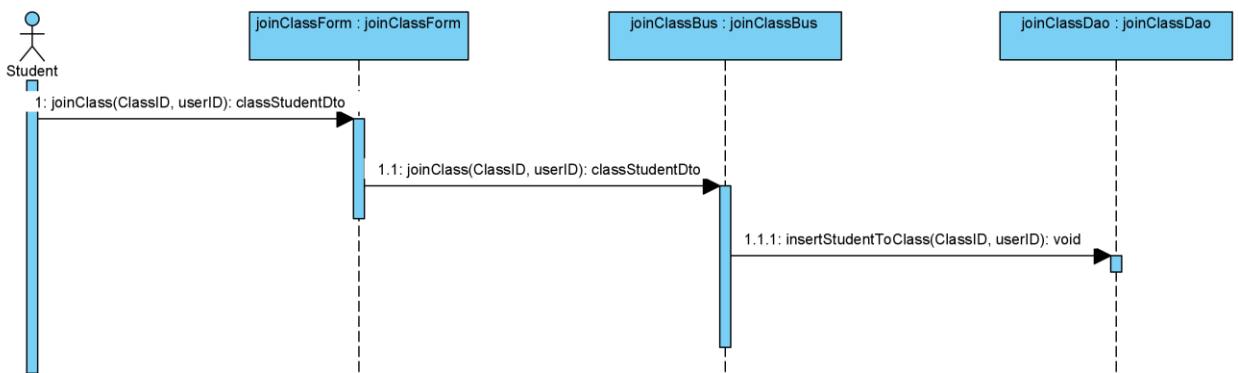


Figure 4. Học sinh tham gia Lớp

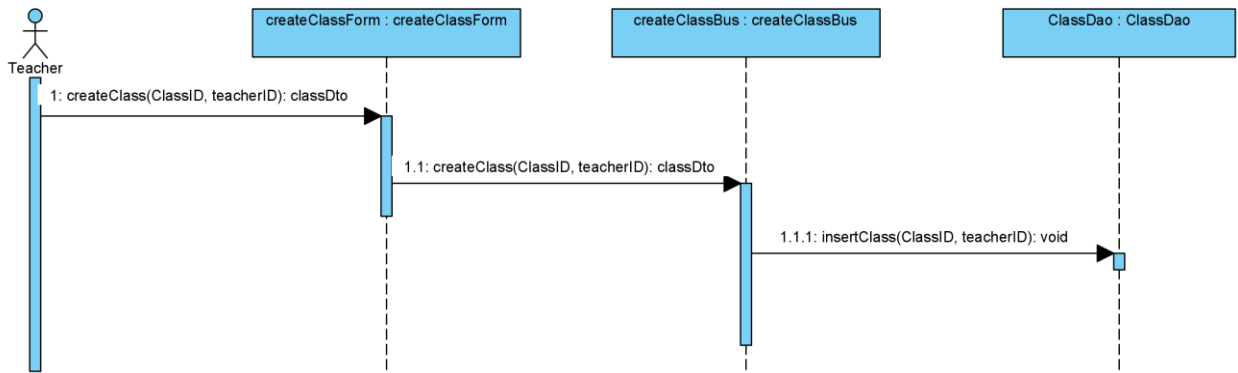


Figure 5. Giáo viên tạo Lớp

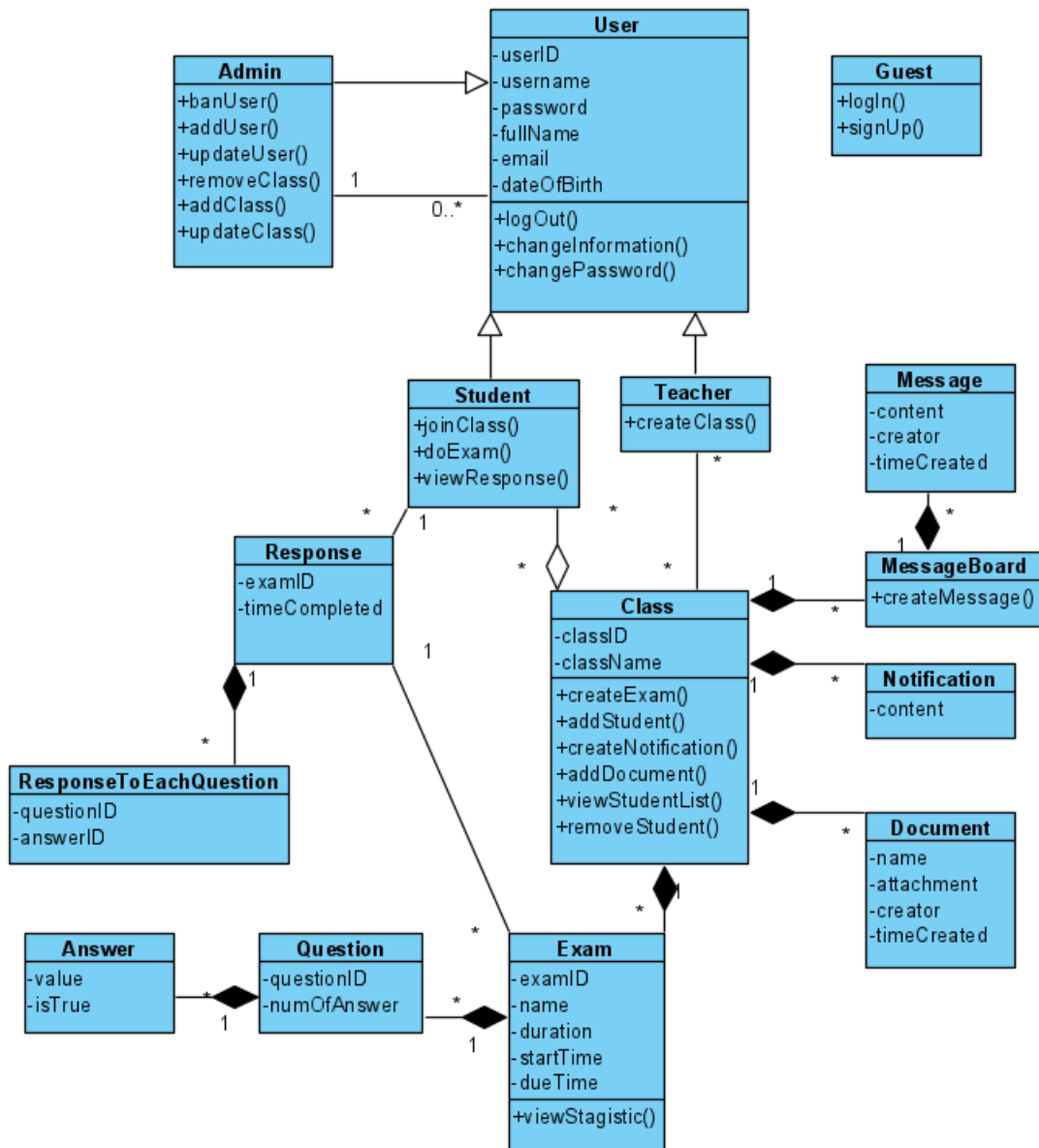
1.7. Yêu cầu phần cứng

ClassPin là một website hoạt động trên trình duyệt web nên yêu cầu phần cứng không quá cao.

- Hoạt động trên trình duyệt web của cả máy tính và các thiết bị thông minh: Chrome, Firefox, Safari, Edge, IE (bản mới)... các trình duyệt có hỗ trợ html, css, javascript và giao thức HTTP.
- Màn hình điều khiển chuột, bàn phím để nhập liệu (có thể sử dụng bàn phím ảo) hoặc màn hình cảm ứng để thao tác với trang web.
- Độ phân giải màn hình tối thiểu 480p, hệ màu 32bit trở lên.
- Đảm bảo dung lượng đĩa trống đủ để lưu cache của trang web khoảng vài mb.
- Băng thông mạng tối thiểu 100 kbps, ổn định.

2. Biểu đồ lớp

2.1. Biểu đồ lớp



2.2. Đặc tả các lớp

2.2.1. Lớp Student

Lớp Student thừa kế từ lớp User.

| STT | Tên thuộc tính | Loại | Ràng buộc | Ý nghĩa |
|-----|----------------|---------|-----------|------------------|
| 1 | Username | public | | Tên người dùng |
| 2 | Password | private | | Mật khẩu |
| 3 | Email | public | | Email student |
| 4 | Phone | public | | Số điện thoại |
| 5 | Avatar | public | | Đường dẫn avatar |

[Danh sách các các phương thức chính]

| STT | Tên phương thức | Loại | Ràng buộc | Ý nghĩa |
|-----|-----------------|-----------|-----------|----------------------|
| 1 | joinClass() | protected | | Tham gia một lớp học |
| 2 | doExam() | private | | Làm đề thi |
| 3 | viewReponse() | private | | Xem kết quả bài thi |

2.2.2. Lớp Class

| STT | Tên thuộc tính | Loại | Ràng buộc | Ý nghĩa |
|-----|----------------|---------|-----------|--------------------------------|
| 1 | classID | private | | Định danh duy nhất của lớp học |
| 2 | className | public | | Tên lớp học |

[Danh sách các các phương thức chính]

| STT | Tên phương thức | Loại | Ràng buộc | Ý nghĩa |
|-----|----------------------|-----------|-----------|--------------------------------------|
| 1 | createExam() | protected | | Tạo bài kiểm tra mới |
| 2 | addStudent() | protected | | Thêm học sinh vào lớp học |
| 3 | createNotification() | protected | | Tạo thông báo đến học sinh trong lớp |
| 4 | addDocument() | protected | | Thêm tài liệu vào lớp học |
| 5 | viewStudentList() | protected | | Xem danh sách học sinh của lớp |

| | | | | |
|---|-----------------|-----------|--|-----------------------|
| 6 | removeStudent() | protected | | Xóa học sinh khỏi lớp |
|---|-----------------|-----------|--|-----------------------|

2.2.3. Lớp Teacher:

Lớp Teacher thừa kế từ lớp User.

| STT | Tên thuộc tính | Loại | Ràng buộc | Ý nghĩa |
|-----|----------------|---------|-----------|------------------|
| 1 | Username | public | | Tên người dùng |
| 2 | Password | private | | Mật khẩu |
| 3 | Email | public | | Email student |
| 4 | Phone | public | | Số điện thoại |
| 5 | Avatar | public | | Đường dẫn avatar |

[Danh sách các các phương thức chính]

| STT | Tên phương thức | Loại | Ràng buộc | Ý nghĩa |
|-----|-----------------|-----------|-----------|-----------------|
| 1 | createClass () | protected | | Tạo một lớp học |

3. Thuật toán và cấu trúc dữ liệu

3.1. Các thuật toán

Hệ thống sử dụng thuật toán phát sinh id lớp học từ module shortid thông qua hàm shortid.generate(). Thuật toán của shortid tạo ra những id không trùng lặp nhờ sử dụng kỹ thuật băm ngẫu nhiên thời gian.

Hệ thống sử dụng Bcrypt để mã hoá mật khẩu. Bcrypt dùng thuật toán băm có tên là Blowfish.

3.2. Các cấu trúc dữ liệu

Cấu trúc dữ liệu được sử dụng là:

- + Array javascript

- + Object javascript

- + ObjectID (MongoDB)

- Khi Search lớp học theo tên sẽ tìm những lớp học có chứa chữ được search, không phân biệt ký tự hoa hay thường.

- Khi tạo user mật khẩu sẽ được kiểm tra có độ dài nhỏ nhất là 8 và độ dài lớn nhất là 30.

- Password được hash bằng thuật toán băm Blowfish.

4. Quản trị dự án và kế hoạch làm việc

4.1. Tổng hợp đóng góp từ các thành viên của nhóm

- Thiết kế kiến trúc và hệ thống:
 - 18120256 – Vũ Lê Tuấn: 25%
 - 18120305 – Lê Quốc Đạt: 25%
 - 18120273 – Phạm Hoàng An: 25%
 - 18120299 – Trương Công Quốc Cường: 25%
- Biểu đồ lớp:
 - 18120256 – Vũ Lê Tuấn: 25%
 - 18120305 – Lê Quốc Đạt: 25%
 - 18120273 – Phạm Hoàng An: 25%
 - 18120299 – Trương Công Quốc Cường: 25%
- Thuật toán và cấu trúc dữ liệu:
 - 18120305 – Lê Quốc Đạt: 50%
 - 18120273 – Phạm Hoàng An: 50%
- Quản trị dự án và kế hoạch làm việc:
 - 18120256 – Vũ Lê Tuấn: 75%
 - 18120305 – Lê Quốc Đạt: 25%

4.2. Báo cáo tiến độ và điều phối dự án

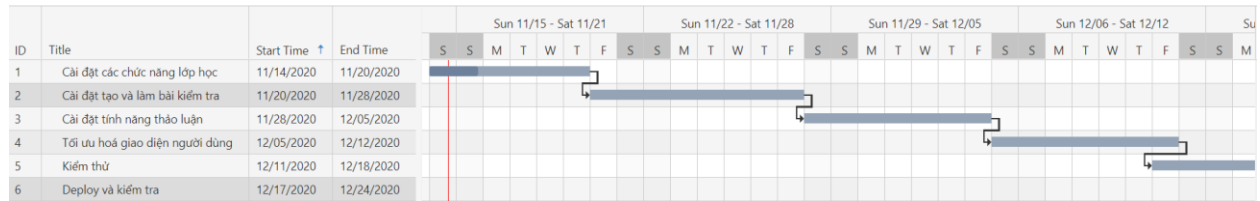
Các use case đã cài đặt:

- Đăng nhập, đăng xuất, đăng kí tài khoản, cập nhật thông tin tài khoản, sửa mật khẩu, ảnh đại diện.
- Tham gia lớp học, truy cập khoá học đã đăng kí.
- Tạo lớp học, truy cập khoá học đã tạo, tìm kiếm lớp học đã tham gia bằng tên.

Các use case đang giải quyết:

- Tạo đề thi.
- Tham gia thi.
- Quản lý lớp học.

4.3. Kế hoạch thực hiện



4.4. Phân rã trách nhiệm (Breakdown of Responsibilities)

| Thành viên | Công việc |
|------------------------|--|
| Vũ Lê Tuấn | Điều phối tích hợp, hệ thống admin, cài đặt các chức năng lớp học (giáo viên) |
| Lê Quốc Đạt | Hệ thống đăng nhập đăng ký (Guess, đăng nhập, đăng xuất, đăng kí, thông tin tài khoản). Cài đặt các chức năng lớp học (học sinh) |
| Phạm Hoàng An | Hệ thống kiểm tra, kiểm thử |
| Trương Công Quốc Cường | Hệ thống thảo luận, thông báo, tài liệu, kiểm thử |

5. Tham khảo

<https://www.npmjs.com/package/shortid>

<https://www.npmjs.com/package/bcrypt>

<https://www.uml-diagrams.org>

<https://www.uml-diagrams.org/component-diagrams.html>

<https://silo.tips/download/web-application-development-using-uml>