

**ĐẠI HỌC BÁCH KHOA HÀ NỘI**  
**TRƯỜNG CÔNG THÔNG TIN VÀ TRUYỀN THÔNG**



**BÁO CÁO NGHIÊN CỨU TÍNH KHẢ THI CỦA DỰ ÁN**

**Môn học: Nhập môn công nghệ phần mềm**

**Giáo viên hướng dẫn:**

**TS Bùi Thị Mai Anh**

**Nhóm sinh viên thực hiện:**

STT	Họ và tên	Mã sinh viên
1	Trần Quang Huy	20226109
2	Nguyễn Lương Hoàng Tùng	20226129
3	Nguyễn Trung Hiếu	20226082
4	Ngô Mạnh Hùng	20226083

**Hà Nội - 2024**

MỤC LỤC

I. TÓM TẮT CHUNG VỀ YÊU CẦU CỦA PROJECT	Error! Bookmark not defined.
II. PHÂN TÍCH YÊU CẦU SƠ BỘ	4
III. QUY TRÌNH PHÁT TRIỂN	8
IV. MỘT SỐ KẾT QUẢ DỰ KIẾN	10
V. TÍNH KHẢ THI VỀ MẶT KỸ THUẬT	12
VI. PHÂN TÍCH RỦI RO	14
VIII. CONCLUSION/ KẾT LUẬN	15

## **I. Tóm tắt chung về dự án**

### **1. User & Client**

Hệ thống HUSTHub được đề xuất thiết kế dành cho các sinh viên, giảng viên và Phòng Đào Tạo ĐHBKHN. Đại học Bách Khoa Hà Nội, đại diện bởi ban quản lý trường, là bên yêu cầu và tài trợ cho dự án này. Họ yêu cầu phát triển hệ thống để cải thiện quản lý các hoạt động hành chính, học tập và giảng dạy của nhà trường.

### **2. Mục đích phát triển**

Việc phát triển hệ thống này giúp tối ưu hóa quy trình quản lý học vụ (quản lý điểm số, quản lý hồ sơ sinh viên, ...), giúp sinh viên, giảng viên, nhân viên hành chính có một hệ thống thống nhất để quản lý thông tin liên quan đến học tập, giảng dạy và các hoạt động khác, ngoài ra cũng nâng cao hiệu quả quản lý hành chính của trường, giảm thiểu các quy trình thủ công, và hạn chế sai sót trong việc xử lý dữ liệu.

### **3. Các nghiệp vụ hỗ trợ**

- Quản lý thông tin
- Quản lý môn học / thời khóa biểu
- Quản lý điểm số
- Quản lý học phí

### **4. Mục tiêu sau khi phát triển**

- **Tối ưu hóa quản lý học vụ:** Hệ thống HUSTHub sẽ giúp tối ưu hóa quy trình quản lý học vụ, đặc biệt trong việc lưu trữ và cập nhật thông tin về sinh viên, điểm số, hồ sơ, và lịch học. Với sự hỗ trợ của hệ thống, các dữ liệu sẽ được xử lý một cách nhanh chóng và chính xác, giảm thiểu tối đa sai sót do các quy trình thủ công. Điều này giúp các bộ phận quản lý tiết kiệm thời gian và nguồn lực, đồng thời tăng cường khả năng quản lý toàn diện thông tin học tập của sinh viên.
- **Đồng bộ hóa thông tin và quy trình:** xây dựng một nền tảng thống nhất, nơi thông tin liên quan đến học tập và giảng dạy được đồng bộ hóa giữa sinh viên, giảng viên và các phòng ban. Thay vì phải sử dụng nhiều hệ thống hoặc công cụ khác nhau, tất cả thông tin sẽ được tập trung trên HUSTHub, từ đó giảm thiểu sự chồng chéo và đảm bảo tính chính xác. Mọi thay đổi hoặc cập nhật đều có thể được đồng bộ tức thời, tạo ra một quy trình làm việc liền mạch và hiệu quả hơn.
- **Nâng cao chất lượng giảng dạy và học tập:** Hệ thống sẽ tạo điều kiện thuận lợi cho giảng viên trong việc quản lý tài liệu học tập, theo dõi tiến độ và đánh giá kết quả học tập của sinh viên. Đồng thời, sinh viên cũng sẽ có một công cụ hiệu quả để theo dõi quá trình học tập của mình, tiếp cận tài liệu và thông tin liên quan một cách dễ dàng.

## **II. PHÂN TÍCH YÊU CẦU SƠ BỘ**

### **1. Tổng quan về ứng dụng**

#### **1.1. Mục tiêu**

Xây dựng một nền tảng quản lý đào tạo toàn diện dành riêng cho Đại học Bách Khoa Hà Nội nhằm tự động hóa và tối ưu hóa các quy trình quản lý đào tạo, bao gồm quản lý sinh viên, giảng viên, chương trình học, lịch học và kết quả học tập. Hệ thống hướng tới việc cải thiện hiệu quả, tính minh bạch, và khả năng truy cập thông tin dễ dàng cho các bên liên quan, đồng thời hỗ trợ việc ra quyết định dựa trên dữ liệu thực tế, từ đó nâng cao chất lượng quản lý và trải nghiệm học tập tại ĐHBKHN.

#### **1.2. Quy trình nghiệp vụ**

Khi thực hiện xây dựng, thiết kế phát triển hệ thống, nhóm thực hiện chia ra làm 4 chức năng chính. Nhóm chia thành 4 quy trình nghiệp vụ khác nhau và đi so sánh quy trình trước đây và nếu có hệ thống quản lí.

##### **1.2.1. Quản lí thông tin**

- *Trước đây*: Thông tin được lưu trữ dưới dạng tài liệu giấy, việc tìm kiếm phải dựa vào mã sinh viên hoặc mã giảng viên và thực hiện thủ công qua các sổ sách. Khi giảng viên hoặc sinh viên cần tra cứu thông tin, họ phải đến bộ phận nhân sự để yêu cầu trích xuất và thường phải chờ đợi để nhận được kết quả.

- *Hiện nay*: Hệ thống sẽ giúp số hóa toàn bộ dữ liệu, cho phép lưu trữ và truy xuất thông tin một cách nhanh chóng, chính xác và dễ dàng. Sinh viên và giảng viên có thể tự tra cứu thông tin trực tuyến mọi lúc, mọi nơi mà không cần phải đến bộ phận hành chính, từ đó giảm thiểu thời gian chờ đợi và tăng tính hiệu quả trong công tác quản lý.

##### **1.2.2. Quản lí thời khoá biểu**

- *Trước đây*: Chỉ có một bảng thời khóa biểu chung cho toàn trường áp dụng cho cả giảng viên và sinh viên. Mỗi giảng viên và sinh viên phải tự ghi chú lại thời khóa biểu của riêng mình để theo dõi, dẫn đến việc dễ xảy ra sai sót hoặc nhầm lẫn.

- *Hiện nay*: Hệ thống sẽ tự động cá nhân hóa thời khóa biểu cho từng giảng viên và sinh viên, giúp họ dễ dàng tra cứu lịch học và lịch dạy trực tuyến. Thời khóa biểu được cập nhật theo thời gian thực, giúp tránh nhầm lẫn, điều chỉnh kịp thời khi có thay đổi, và tạo sự thuận tiện trong việc theo dõi lịch trình cá nhân mà không cần phải ghi chép thủ công.

##### **1.2.3. Quản lí điểm số**

- *Trước đây*: Giảng viên dựa vào danh sách sinh viên của lớp để chấm bài và điền điểm thủ công vào danh sách giấy. Sau đó, điểm số được dán lên bảng thông tin để sinh viên tự đến tra cứu và nhận kết quả (FAMI). Quá trình này mất thời gian và dễ gây ra sai sót trong việc nhập điểm.

- *Hiện nay*: Hệ thống sẽ cho phép giảng viên nhập điểm trực tiếp lên nền tảng trực tuyến, từ đó tự động cập nhật vào hệ thống quản lý của sinh viên. Sinh viên có thể tra cứu điểm số cá nhân của mình ngay lập tức thông qua hệ thống mà không cần phải đến

xem bảng thông tin, giúp tiết kiệm thời gian, đảm bảo tính chính xác và minh bạch trong quá trình công bố kết quả.

#### **1.2.4. Quản lý học phí**

- *Trước đây*: Thông tin về học phí của sinh viên được lưu trữ trên giấy tờ hoặc các file Excel riêng lẻ, việc theo dõi và quản lý học phí phải thực hiện thủ công, dẫn đến khó khăn trong việc kiểm tra, cập nhật và đối soát thông tin. Sinh viên phải đến trực tiếp để nộp học phí và xác nhận trạng thái thanh toán.

- *Hiện nay*: Hệ thống sẽ tự động hóa việc quản lý học phí, cho phép sinh viên tra cứu và thanh toán học phí trực tuyến. Thông tin về học phí được cập nhật theo thời gian thực, giúp sinh viên dễ dàng theo dõi trạng thái thanh toán của mình, đồng thời nhà trường có thể quản lý và đối soát dữ liệu một cách nhanh chóng, chính xác và minh bạch hơn.

### **1.3. Vai trò và trách nhiệm người dùng**

Hệ thống sẽ có 3 loại user chính:

- **Admin:**
  - *Quyền hạn*: Tất cả các quyền quản lý và điều hành dự án.
  - *Trách nhiệm*:
    - Quản lý toàn bộ hồ sơ của giảng viên và sinh viên.
    - Quản lý thời khóa biểu, bao gồm tạo và phân lịch cho các môn học và tín chỉ.
    - Quản lý điểm số, bao gồm việc điều chỉnh và xác nhận điểm.
    - Nhập và quản lý thông tin học phí, bao gồm theo dõi và đối soát dữ liệu thanh toán.
- **Giảng viên:**
  - *Quyền hạn*:
    - Tự cập nhật thông tin cá nhân của mình trên hệ thống.
    - Quản lý hồ sơ của sinh viên trong lớp mà mình chủ nhiệm, bao gồm việc theo dõi tiến độ học tập và điểm số.
    - Xem thời khóa biểu cá nhân và của lớp học mình giảng dạy.
    - Nhập điểm cho sinh viên, chỉnh sửa và cập nhật điểm khi cần thiết.
- **Sinh viên:**
  - *Quyền hạn*:
    - Tự cập nhật thông tin cá nhân của mình.
    - Xem thời khóa biểu cá nhân.
    - Tra cứu điểm số của bản thân trong từng môn học.
    - Theo dõi và tra cứu thông tin về học phí, bao gồm trạng thái thanh toán và các khoản

### **1.4. Tương tác với hệ thống khác**

Hệ thống sẽ có khả năng tương tác với một số nền tảng bên ngoài để nâng cao trải nghiệm của người dùng. Cụ thể:

- **Google Maps:** Hệ thống sẽ tích hợp Google Maps để hỗ trợ hiển thị bản đồ, giúp người dùng dễ dàng tìm kiếm địa điểm trong và xung quanh trường, bao gồm các tòa nhà, phòng học và khu vực liên quan.

- **Thanh toán trực tuyến:** Hệ thống sẽ tích hợp với các cổng thanh toán trực tuyến để sinh viên có thể thanh toán học phí và các chi phí khác một cách nhanh chóng và thuận tiện.

- **YouTube:** Hệ thống có thể liên kết với YouTube để hỗ trợ học trực tuyến qua các video hướng dẫn, bài giảng, và nội dung học tập minh họa, tạo điều kiện cho việc học tập từ xa.

- **Google Photos:** Hệ thống sẽ tích hợp Google Photos để sử dụng các hình ảnh minh họa trong quá trình giảng dạy, học tập hoặc trình bày thông tin trực quan cho các nội dung học thuật.

## 1.5. Những cân nhắc khi triển khai hệ thống

Việc thiết kế và phát triển hệ thống được thực hiện theo từng mốc milestones cụ thể nhằm đảm bảo tiến độ và chất lượng của dự án. Tuy nhiên, do những hạn chế về ngân sách và công nghệ hiện tại, nhóm phát triển chỉ có khả năng triển khai hệ thống trên môi trường cục bộ (local). Trong tương lai, nếu có sự hỗ trợ về nguồn lực và kinh phí, nhóm có thể mở rộng dự án và triển khai hệ thống trên máy chủ thật, cho phép sử dụng quy mô lớn hơn và truy cập từ xa.

Ngoài ra, hệ thống được thiết kế với mục tiêu thân thiện với người dùng, giúp họ có thể thao tác dễ dàng. Các hướng dẫn sử dụng chi tiết sẽ được cung cấp để người dùng có thể nắm bắt cách vận hành hệ thống một cách nhanh chóng. Người dùng chỉ cần đọc tài liệu hướng dẫn và làm theo các bước đơn giản để sử dụng các tính năng của hệ thống một cách hiệu quả. Điều này không chỉ giảm thiểu thời gian đào tạo mà còn tạo điều kiện thuận lợi cho việc mở rộng và áp dụng hệ thống vào thực tiễn trong trường hợp triển khai trên quy mô lớn hơn.

## 2. Yêu cầu chức năng:

### 2.1. Tuyên bố về chức năng:

- **Chức năng Quản lý thông tin:**

Chức năng này cho phép người dùng cập nhật, lưu trữ và quản lý các thông tin cá nhân của sinh viên, bao gồm họ tên, ngày sinh, địa chỉ, thông tin liên hệ và các dữ liệu liên quan khác. Mọi thay đổi hoặc cập nhật về thông tin cá nhân đều được hệ thống ghi nhận và bảo mật.

- **Chức năng quản lý điểm số**

Chức năng này hỗ trợ việc nhập, theo dõi và quản lý điểm số của sinh viên cho các môn học trong suốt quá trình học tập. Người dùng có thể tra cứu, xuất báo cáo và thực

hiện các điều chỉnh cần thiết đối với điểm số của sinh viên dựa trên kết quả học tập và đánh giá từ giáo viên.

- ***Chức năng Quản lý thời khóa biểu***

Chức năng này cho phép tạo lập, cập nhật và theo dõi thời khóa biểu của các lớp học và sinh viên. Hệ thống sẽ cung cấp thông tin về lịch học chi tiết cho từng lớp, từng môn học, giúp quản lý thời gian học tập hiệu quả và tránh xung đột lịch học.

- ***Chức năng Quản lý học phí***

Chức năng này cho phép theo dõi và quản lý các khoản học phí của sinh viên. Hệ thống sẽ hỗ trợ việc lập hóa đơn, ghi nhận thanh toán, theo dõi tình trạng học phí, và cảnh báo khi có khoản nợ chưa thanh toán. Điều này giúp đảm bảo minh bạch và chính xác trong quản lý tài chính học phí.

## **2.2. Khả năng bảo mật và người dùng**

Hệ thống quản lý trường học hỗ trợ ba loại người dùng: Giảng viên, Sinh viên, và Quản trị viên. Để truy cập vào hệ thống, tất cả người dùng cần đăng nhập bằng tên tài khoản và mật khẩu. Mỗi đối tượng người dùng sẽ có các quyền hạn khác nhau dựa trên vai trò của mình trong hệ thống.

- ***Giảng viên***

Giảng viên có quyền truy cập và quản lý thông tin liên quan đến các lớp học mà họ giảng dạy. Điều này bao gồm việc cập nhật điểm số, quản lý thời khóa biểu của các lớp học, và gửi thông báo hoặc tài liệu học tập cho sinh viên. Tuy nhiên, giảng viên không có quyền thêm, xóa hoặc chỉnh sửa các lớp học mà họ không quản lý, quyền này thuộc về quản trị viên.

- ***Sinh viên***

Sinh viên có quyền truy cập để xem thông tin cá nhân, điểm số, thời khóa biểu và tình trạng học phí. Sinh viên có thể đăng ký môn học, cập nhật thông tin cá nhân, và yêu cầu các giao dịch liên quan đến quá trình học tập. Tuy nhiên, sinh viên không có quyền chỉnh sửa thông tin hệ thống như lớp học hoặc điểm số mà chỉ có thể thực hiện các thao tác như xem và đăng ký.

- ***Quản trị viên***

Quản trị viên có toàn quyền quản lý hệ thống, bao gồm thêm, xóa và chỉnh sửa tài khoản của giảng viên và sinh viên. Ngoài ra, họ cũng có thể thêm hoặc xóa các lớp học, khóa học, quản lý thời khóa biểu, và theo dõi các giao dịch học phí. Quản trị viên chịu trách nhiệm đảm bảo rằng hệ thống hoạt động đúng quy trình và tất cả dữ liệu được quản lý chặt chẽ.

Hệ thống cũng hỗ trợ dịch vụ đăng ký khóa học và thời khóa biểu thông qua biểu mẫu trực tuyến. Các biện pháp bảo mật cơ bản như chống HTML/SQL injection được áp dụng để bảo vệ thông tin người dùng, tuy nhiên hệ thống sẽ không bao gồm các dịch vụ bảo vệ phức tạp hơn như chống tấn công DOS/DDOS.

Nhờ phân chia rõ ràng các quyền hạn cho từng đối tượng người dùng, hệ thống đảm bảo tính bảo mật, hiệu quả và chính xác trong việc quản lý các hoạt động liên quan đến trường học.

### 2.3. Tính năng tùy chọn

Đây là những tính năng mà người dùng có thể chưa nghĩ đến hoặc chưa đề xuất, nhưng đội ngũ phát triển sẽ gợi ý dựa trên kinh nghiệm của họ nhằm cải thiện trải nghiệm sử dụng hệ thống quản lý trường học.

- ***Tìm kiếm thông tin sinh viên hoặc giảng viên:*** Đội ngũ phát triển đề xuất một tính năng cho phép người dùng nhập từ khóa hoặc thông tin cụ thể (như họ tên, mã số sinh viên giảng viên) để nhanh chóng tìm kiếm trong cơ sở dữ liệu về sinh viên, giảng viên hoặc các lớp học. Tính năng này giúp dễ dàng tra cứu hồ sơ cá nhân, kết quả học tập, hoặc thời khóa biểu mà không cần duyệt qua toàn bộ hệ thống.

- ***Theo dõi tiến trình học tập:*** Gợi ý thêm một tính năng cho phép sinh viên và giảng viên theo dõi tiến trình học tập hoặc giảng dạy. Tính năng này có thể bao gồm đồ thị hiển thị sự tiến bộ của sinh viên qua các kỳ học, giúp giảng viên đánh giá và sinh viên tự theo dõi sự phát triển của bản thân.

### 2.4. Phạm vi dự án

Hệ thống mang tính chất quản lý và hỗ trợ giảng viên, giáo viên trong các công tác xoay quanh việc quản lý, dạy học trong trường Đại học. Do thời gian gấp rút trong phạm vi 1 kì học, dự án chỉ đưa ra các chức năng ở mức độ cơ bản nhất. Để dự án được đưa vào thực tế, chắc chắn sẽ còn nhiều vấn đề như: thanh toán qua các hệ thống khác, các chức năng cao hơn như quản lý từng sinh viên, thông báo, cập nhật thông tin... Đồng thời các trường thông tin cơ bản sẽ không thể thay đổi (ví dụ như định dạng mã sinh viên, định dạng tín chỉ,...), các quyền quản trị của admin chưa được hoàn thiện quá nhiều.

## III. QUY TRÌNH PHÁT TRIỂN

### 1. Mô hình phát triển

Nhóm đã lựa chọn **Mô hình thác nước cải tiến (Modified Waterfall)** để phát triển phần mềm. Đây là phiên bản biến thể của mô hình thác nước truyền thống, trong đó quy trình vẫn được chia thành các giai đoạn nhưng có bổ sung các vòng lặp phản hồi, cho phép quay lại các giai đoạn trước nếu cần. Phương pháp linh hoạt này phù hợp cho các dự án cần sự cấu trúc nhưng cũng yêu cầu khả năng điều chỉnh và tinh chỉnh trong quá trình thực hiện.

### 2. Lý do lựa chọn

- ***Cách tiếp cận có cấu trúc:*** Mô hình thác nước cung cấp một luồng làm việc rõ ràng và tuần tự, giúp đảm bảo rằng mỗi giai đoạn được hoàn thành trước khi chuyển



sang giai đoạn tiếp theo. Điều này rất quan trọng để duy trì sự tập trung và đảm bảo không bỏ sót bất kỳ bước nào.

- **Tính linh hoạt trong việc lặp lại:** Trong mô hình thác nước truyền thống, mỗi giai đoạn phải hoàn thành trước khi chuyển sang giai đoạn tiếp theo. Tuy nhiên, phiên bản cải tiến cho phép nhóm quay lại các giai đoạn trước khi có vấn đề phát sinh sau này. Điều này đặc biệt hữu ích trong các dự án thực tế, nơi có thể xuất hiện những thách thức hoặc yêu cầu thay đổi không thể lường trước.

- **Kiểm soát dự án và tài liệu:** Mô hình thác nước cải tiến cho phép kiểm soát dự án tốt hơn thông qua việc tài liệu hoá kỹ lưỡng ở mỗi giai đoạn. Điều này đảm bảo rằng nhóm có thể theo dõi tiến độ, chia sẻ kiến thức.

- **Giảm thiểu rủi ro:** Khả năng quay lại và sửa chữa các vấn đề giúp giảm rủi ro, tăng khả năng hoàn thành sản phẩm chất lượng đúng thời hạn.

### 3. Process Outline

#### 3.1. Lên ý tưởng và báo cáo tính khả thi (24/09-08/10)

Giai đoạn đầu tiên là quá trình lên ý tưởng và báo cáo tính khả thi. Nhóm dự án sẽ tổ chức các buổi thảo luận nội bộ để xây dựng ý tưởng về hệ thống quản lý học vụ HUSTHub, dựa trên nhu cầu thực tế của sinh viên, giảng viên, và Phòng Đào Tạo. Sau khi xác định được các tính năng cốt lõi và giá trị mà hệ thống sẽ mang lại, nhóm sẽ lập một báo cáo tính khả thi (feasibility report). Báo cáo này sẽ đánh giá các yếu tố như khả năng phát triển, công nghệ phù hợp, tài nguyên cần thiết, và những rủi ro có thể gặp phải trong quá trình thực hiện dự án. Mục tiêu của giai đoạn này là xác định tính khả thi tổng thể của dự án và chuẩn bị cho các bước phát triển tiếp theo.

#### 3.2. Khảo sát và phân tích yêu cầu (09/10 - 22/10)

Nhóm sẽ phân tích các tài liệu tham khảo từ Đại học Bách Khoa Hà Nội, các hệ thống quản lý học vụ hiện có. Sau khi nghiên cứu, các yêu cầu sẽ được tổng hợp lại để tạo thành một tài liệu phân tích yêu cầu chi tiết (Software Requirement Specification - SRS). Tài liệu này sẽ liệt kê đầy đủ các chức năng, đặc điểm kỹ thuật và yêu cầu phi chức năng của hệ thống. Giai đoạn này sẽ giúp nhóm xác định rõ phạm vi dự án và làm cơ sở cho các giai đoạn thiết kế và phát triển tiếp theo.

#### 3.3. Thiết kế hệ thống (23/10 - 05/11)

Trong giai đoạn này, nhóm sẽ chuyển từ giai đoạn ý tưởng và phân tích yêu cầu sang việc thiết kế hệ thống cụ thể. Đầu tiên là thiết kế kiến trúc hệ thống, bao gồm việc xác định cấu trúc phần mềm, thiết kế cơ sở dữ liệu, và lựa chọn công nghệ phù hợp. Sau khi kiến trúc hệ thống được xác định, nhóm sẽ tiến hành thiết kế giao diện người dùng (UI/UX) nhằm đảm bảo hệ thống dễ sử dụng và thân thiện với sinh viên, giảng viên và nhân viên hành chính. Các giao diện quan trọng như trang quản lý điểm số, lịch học và hệ thống thông báo sẽ được tạo dưới dạng mockup để thu thập phản hồi từ các bên liên quan, trước khi chuyển sang giai đoạn phát triển.

#### 3.4. Phát triển và lập trình (06/11 - 03/12)

Giai đoạn phát triển hệ thống sẽ bắt đầu với việc lập trình các chức năng backend. Nhóm sẽ xây dựng các module chính như quản lý điểm số, hồ sơ sinh viên, và hệ thống quản lý lịch học. Các API cần thiết sẽ được phát triển để kết nối giữa các module và cơ sở dữ liệu. Sau khi hoàn thiện backend, nhóm sẽ tiến hành phát triển giao diện người dùng dựa trên thiết kế đã được phê duyệt. Trong quá trình này, nhóm sẽ kiểm tra từng module và giao diện để đảm bảo chúng hoạt động mượt mà và tương thích với nhau. Giai đoạn này cũng sẽ bao gồm việc liên kết frontend với backend, tạo ra một hệ thống hoàn chỉnh và sẵn sàng cho các bước kiểm thử.

### **3.5. Kiểm thử hệ thống (04/12-10/12)**

Trong giai đoạn này, hệ thống sẽ được kiểm thử kỹ lưỡng nhằm đảm bảo rằng tất cả các chức năng hoạt động chính xác và đúng với yêu cầu ban đầu. Nhóm sẽ bắt đầu với kiểm thử từng module riêng lẻ (unit testing), sau đó tiến hành kiểm tra tích hợp giữa các module (integration testing) để đảm bảo các phần của hệ thống kết nối với nhau một cách liền mạch. Sau khi hoàn tất kiểm thử chức năng, nhóm sẽ thực hiện kiểm thử toàn bộ hệ thống (system testing), bao gồm kiểm tra giao diện và trải nghiệm người dùng để đảm bảo hệ thống hoạt động ổn định và dễ sử dụng. Nếu phát hiện lỗi hoặc sự cố, nhóm sẽ điều chỉnh và khắc phục trước khi chuyển sang giai đoạn triển khai.

### **3.6. Triển khai (11/12-17/12)**

Sau khi hệ thống đã vượt qua các bài kiểm thử, nhóm sẽ tiến hành triển khai hệ thống trên web (nếu có).

Giai đoạn cuối cùng là bảo trì và hỗ trợ sau khi hệ thống đi vào hoạt động chưa thực hiện ở dự án này do quy mô và tính chất đồ án môn học.

## **IV. MỘT SỐ KẾT QUẢ DỰ KIẾN**

Dự án hệ thống quản lý trường học nhằm cung cấp một giải pháp toàn diện cho việc quản lý thông tin, học tập và hành chính của nhà trường, thông qua việc phát triển các module chức năng chi tiết. Các giai đoạn sẽ kèm theo tài liệu cần thiết và mô tả chức năng cụ thể để đảm bảo hệ thống đáp ứng yêu cầu của khách hàng

### **1. Giai đoạn Thu thập yêu cầu:**

#### ***Mục tiêu:***

- Xác định rõ ràng các yêu cầu chức năng và phi chức năng của khách hàng.
- Phỏng vấn và thu thập yêu cầu từ các bên liên quan: giảng viên, sinh viên, quản trị viên.

#### ***Tài liệu cần thiết:***

- Tài liệu Yêu cầu Phần mềm (SRS - Software Requirements Specification): Mô tả chi tiết tất cả các yêu cầu mà hệ thống cần đáp ứng, bao gồm cả yêu cầu chức năng và phi chức năng.

#### ***Chức năng dự kiến:***

- Xác định các chức năng cơ bản của hệ thống như quản lý hồ sơ giảng viên, sinh viên, quản lý thời khóa biểu, điểm số và học phí.

***Ví dụ cụ thể:***

- Hệ thống cho phép sinh viên tự nhập và chỉnh sửa thông tin cá nhân.
- Hệ thống quản lý thời khóa biểu cho từng sinh viên và giảng viên theo lớp học, môn học.
- Hệ thống hỗ trợ tìm kiếm thông tin sinh viên/giảng viên dựa trên mã số hoặc tên.

## **2. Giai đoạn Phân tích hệ thống**

***Mục tiêu:***

- Phân tích yêu cầu từ SRS và xác định cách thức triển khai hệ thống.
- Xây dựng các mô hình dữ liệu và luồng thông tin trong hệ thống.

***Tài liệu cần thiết:***

- Báo cáo Phân tích hệ thống (System Analysis Report): Phân tích chi tiết các thành phần của hệ thống, bao gồm sơ đồ luồng dữ liệu (DFD), sơ đồ use-case và các trường hợp sử dụng.

***Chức năng dự kiến:***

- Xây dựng sơ đồ cơ sở dữ liệu cho các đối tượng chính như: sinh viên, giảng viên, lớp học, môn học, điểm số, học phí.
- Phân tích các mối quan hệ giữa các thành phần trong hệ thống:
  - o Mỗi sinh viên có một danh sách môn học, thời khóa biểu, và điểm số riêng.
  - o Giảng viên có quyền truy cập để xem và nhập điểm của sinh viên trong các lớp mà họ dạy.
  - o Hệ thống quản lý học phí theo học kỳ và theo từng sinh viên.

## **3. Giai đoạn Thiết kế hệ thống**

***Mục tiêu:***

- Thiết kế kiến trúc chi tiết của hệ thống, bao gồm giao diện người dùng (UI) và cơ sở dữ liệu.
- Đảm bảo rằng hệ thống có thể mở rộng và bảo mật.

***Tài liệu cần thiết:***

- Tài liệu Thiết kế Hệ thống (SDD - System Design Document): Mô tả chi tiết về kiến trúc hệ thống, bao gồm thiết kế cơ sở dữ liệu, thiết kế giao diện người dùng và các giao thức bảo mật.

***Chức năng dự kiến:***

- Thiết kế giao diện người dùng thân thiện cho các loại người dùng (Admin, Giảng viên, Sinh viên).
  - o Giao diện quản lý hồ sơ sinh viên/giảng viên với các trường thông tin cơ bản như tên, mã số, ngày sinh, lớp học.
  - o Giao diện nhập và xem thời khóa biểu cá nhân.

- Giao diện nhập điểm và xem điểm của sinh viên theo từng môn học.
- Thiết kế cơ sở dữ liệu đảm bảo lưu trữ an toàn và nhanh chóng các thông tin lớn của nhà trường.

#### 4. Giai đoạn Phát triển

##### **Mục tiêu:**

- Phát triển các module chức năng theo thiết kế đã được phê duyệt.
- Đảm bảo rằng hệ thống được phát triển tuân thủ các yêu cầu và tiêu chuẩn đã đặt ra.

##### **Tài liệu cần thiết:**

- Tài liệu Mã nguồn (Source Code Documentation): Ghi lại cấu trúc mã nguồn, các lớp và phương thức được sử dụng.

##### **Chức năng dự kiến:**

- Phát triển các chức năng chính:
  - Module quản lý sinh viên: Cho phép nhập, chỉnh sửa và xóa thông tin sinh viên.
  - Module quản lý giảng viên: Quản lý thông tin giảng viên và các lớp học.
  - Module quản lý điểm số: Cho phép giảng viên nhập điểm cho sinh viên.
  - Module quản lý học phí: Giúp quản trị viên theo dõi và tính toán học phí của từng sinh viên.

#### 5. Giai đoạn Tích hợp và Kiểm thử

##### **Mục tiêu:**

- Tích hợp các module đã phát triển và tiến hành kiểm thử toàn hệ thống.
- Đảm bảo hệ thống hoạt động trơn tru và đáp ứng các yêu cầu đặt ra.

##### **Chức năng dự kiến:**

- Tích hợp các chức năng quản lý sinh viên, giảng viên, điểm số, và học phí để kiểm thử.
- Tiến hành kiểm thử chức năng:
  - Xác minh chức năng nhập, sửa và xóa thông tin sinh viên.
  - Kiểm tra chức năng quản lý điểm số và học phí.
  - Đảm bảo rằng quyền truy cập của các user (Admin, Giảng viên, Sinh viên) hoạt động đúng theo phân quyền.

## **V. TÍNH KHẢ THI VỀ MẶT KỸ THUẬT**

### **1. Yêu cầu chức năng**

- **Quản lý hồ sơ sinh viên và giảng viên:** Hệ thống cho phép tạo, chỉnh sửa và quản lý thông tin cá nhân của sinh viên và giảng viên
- **Quản lý thời khóa biểu:** Sinh viên và giảng viên có thể xem và quản lý thời khóa biểu cá nhân.
- **Quản lý điểm số:** Giảng viên nhập và chỉnh sửa điểm của sinh viên, trong khi sinh viên có thể truy cập và xem kết quả học tập của mình.

- **Quản lý học phí:** Sinh viên có thể theo dõi thông tin học phí và trạng thái thanh toán.
- **Phân quyền người dùng:** Phân quyền người dùng dựa trên vai trò (quản trị viên, giảng viên, sinh viên), mỗi loại người dùng có các quyền hạn khác nhau.

## 2. Yêu cầu phi chức năng

Các yêu cầu phi chức năng bao gồm:

- **Hiệu suất:** Hệ thống phải chạy mượt mà trên localhost và có khả năng xử lý lượng dữ liệu lớn mà không gặp vấn đề về hiệu suất.
- **Bảo mật:** Dù hệ thống không được triển khai trên môi trường công cộng, vẫn cần đảm bảo tính bảo mật nội bộ, ví dụ thông qua xác thực người dùng.
- **Tính thân thiện với người dùng:** Giao diện người dùng phải trực quan và dễ sử dụng cho cả giảng viên, sinh viên và quản trị viên.

## 3. Phương pháp kỹ thuật

### 3.1. Cơ sở dữ liệu

Sử dụng cơ sở dữ liệu quan hệ: SQL Server/MySQL/PostgreSQL, phù hợp với việc lưu trữ dữ liệu và dễ dàng cài đặt trên localhost. Cơ sở dữ liệu này sẽ lưu trữ thông tin về sinh viên, giảng viên, điểm số và học phí.

### 3.2. Phát triển phần mềm

- **Frontend:** Hệ thống sẽ sử dụng HTML, CSS, Javascript cho giao diện người dùng, đảm bảo khả năng phản hồi nhanh và tương thích với các thiết bị khác nhau.
- **Backend:** Sử dụng NodeJS và ExpressJS để quản lý các yêu cầu từ người dùng và tương tác với cơ sở dữ liệu trên localhost.
- **Bảo mật:** Sử dụng các biện pháp xác thực cơ bản để bảo vệ quyền truy cập vào các chức năng quản lý quan trọng.

## 4. Tính khả thi về nguồn lực

### 4.1. Nhân sự dự kiến

Để đảm bảo mỗi thành viên đều được tham gia, trải qua các giai đoạn phát triển dự án, nhóm chia từng module cho các thành viên. Trong đó, mỗi module yêu cầu thiết kế frontend, backend nhằm cải thiện kỹ năng.

Sau các giai đoạn, các thành viên kiểm thử chéo lẫn nhau để đảm bảo hệ thống hoạt động ổn định trên local host.

### 4.2. Thiết bị và môi trường phát triển

- Sử dụng môi trường quản lý version Github.
- Hệ thống được phát triển trên localhost

## 5. Kết luận

Với các kỹ thuật hiện tại và yêu cầu của dự án, hệ thống quản lý trường học có tính khả thi cao khi chạy trên localhost. Hệ thống có thể đáp ứng đầy đủ các yêu cầu chức năng và phi chức năng của người dùng, đồng thời dễ dàng mở rộng trong tương lai nếu cần thiết.

## VI. PHÂN TÍCH RỦI RO

Trong quá trình phát triển sản phẩm, nhóm bọn dự án nhận thấy có thể gặp phải một số rủi ro như sau:

### 1. Rủi ro về thời gian

- **Chậm tiến độ:** Vì nhóm có sử dụng mô hình WaterFall (chia thành nhiều giai đoạn) nên nếu 1 giai đoạn bị chậm thì sẽ ảnh hưởng cả quá trình
- **Thay đổi yêu cầu:** Trong quá trình phát triển, đôi khi nhóm dự án sẽ cần thực hiện những yêu cầu mới để phù hợp với chương trình vì vậy có khả năng dự án sẽ bị kéo dài
- **Kiểm thử sản phẩm:** Không có đủ thời gian kiểm thử sản phẩm do có thể sẽ bị rút ngắn.

### 2. Rủi ro về nguồn lực

- **Thiếu kỹ năng:** Nhóm sinh viên tự nhận thấy còn chưa quá thành thạo những ngôn ngữ để thiết kế Front-end và cả những ngôn ngữ lập trình để xây dựng Back-end giúp web vận hành trơn tru.
- **Thiếu nhân lực:** Vì sản phẩm có khá nhiều chức năng, nên 1 thành viên phải đảm nhận nhiều nhiệm vụ. Nếu có 1 thành viên gặp vấn đề gì đó không thực hiện tiếp dự án thì sẽ bị thiếu nhân lực.

### 3. Rủi ro về chức năng của hệ thống

- **Chức năng của hệ thống chưa đáp ứng đúng nhu cầu:** Trong quá trình phát triển sản phẩm, một vài yêu cầu đã xác định ban đầu đôi khi sẽ cần phải sửa đổi lại để phù hợp hơn với thực tế.
- **Hiệu suất kém:** Hệ thống có thể đáp ứng cho 1 vài users thử nhưng sẽ gặp vấn đề nếu số lượng users thực tế lớn.

Đây là những rủi ro mà nhóm nhận thấy và cần phải chuẩn bị thật kỹ để có các giải pháp giúp vượt qua nhằm đảm bảo sự thành công của dự án.

### 4. Các giải pháp và giải pháp dự trù

- **Giải pháp**
  - o *Đối với rủi ro về thời gian:*
    - Lập kế hoạch chi tiết cho từng giai đoạn, nhóm trưởng sẽ kiểm tra tiến độ
    - Linh hoạt điều chỉnh khi có sự việc nằm ngoài dự tính mà vẫn đảm bảo deadline

- *Đối với rủi ro về nguồn lực:*
  - Mỗi thành viên hỗ trợ thành viên khác các phần là điểm mạnh của bản thân.
  - Vừa làm kết hợp với vừa học những kiến thức mới để áp dụng.
- *Đối với rủi ro về chức năng hệ thống:*
  - Cần xác định rõ, cụ thể chức năng của hệ thống tránh việc thay đổi khi dự án đã bước vào giai đoạn thực thi code.
  - Kiểm thử sớm và thường xuyên để kịp thời sửa lỗi.

- ***Giải pháp dự trù:***

Trong quá trình làm việc sẽ không tránh khỏi những sai lầm vì vậy nhóm cần có kế hoạch dự phòng.

- Linh hoạt điều chỉnh khi có sự việc nằm ngoài dự tính mà vẫn đảm bảo deadline
- Loại bỏ những yêu cầu không cần thiết khiến dự án bị kéo dài
- Nếu đối với lượng dữ liệu lớn mà hệ thống không đáp ứng được nhu cầu thì cần phải cải tiến cơ sở dữ liệu và truy vấn.

## **VII. CONCLUSION/ KẾT LUẬN**

Qua quá trình phân tích tính khả thi, nhóm sinh viên nhận thấy dự án HustHub có tính khả thi về mặt kỹ thuật và tài nguyên. Với thời hạn cho phép, nhóm tin rằng sẽ hoàn thiện được dự án và đáp ứng được yêu cầu của môn học. Tuy nhiên, nhóm dự án sẽ cần lưu ý về các rủi ro tiềm ẩn liên quan đến kiểm thử, bảo mật, và thời gian thực hiện để đảm bảo dự án thành công.