

DẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH



CÔNG NGHỆ PHẦN MỀM (CO3001)

Báo cáo bài tập lớn

BKArch: Hệ thống hỗ trợ chương trình Tutor/Mentor

Giảng viên hướng dẫn: Mai Đức Trung

STT	Họ tên SV	MSSV	Lớp
1	Nguyễn Đức Nghĩa	2312266	L03
2	Phan Huy Quang Minh	2312105	L03
3	Ngô Thanh Hùng	2311289	L03
4	Nguyễn Dắc Nghĩa	2312265	L03
5	Hà Cao Đức Minh	2312059	L03
6	Nguyễn Trí Hải	2314047	L02
7	Trần Tiến Khải	2311566	L03

Thành phố Hồ Chí Minh, tháng 10 năm 2025



Nội dung

Danh sách thành viên và khối lượng công việc	3
1 Tổng quan về dự án	4
1.1 Bối cảnh và tính cấp thiết	4
1.2 Mục đích và mục tiêu dự án	4
1.2.1 Mục đích dự án:	4
1.2.2 Mục tiêu dự án:	5
1.3 Phạm vi dự án	5
1.3.1 Chức năng cốt lõi	5
1.3.2 Chức năng mở rộng (đề xuất)	6
1.4 Định hướng kỹ thuật	6
1.4.1 Mô hình Server-Client	6
1.4.2 ReactJS	8
1.4.3 ExpressJS	8
1.4.4 Git và GitHub	8
1.5 Định hướng mở rộng và nâng cấp	9
1.5.1 Tích hợp AI	9
1.5.2 Xây dựng các tính năng cộng đồng và các diễn đàn:	9
1.5.3 Chương trình học thuật và phi học thuật:	10
1.5.4 Cá nhân hoá học tập:	10
2 Các bên liên quan (Relevant Stakeholders)	11
3 Yêu cầu chức năng (Functional Requirements)	11
3.1 Quản lý người dùng & xác thực	11
3.2 Quản lý tutor & mentee	11
3.3 Quản lý lịch & buổi tư vấn	12
3.4 Dánh giá & phản hồi	12
3.5 Quản lý học liệu & tài nguyên	12
3.6 Chức năng mở rộng (đề xuất)	12
4 Yêu cầu phi chức năng (Non-Functional Requirements)	12
4.1 Hiệu năng (Performance)	12
4.2 Bảo mật (Security)	13
4.3 Khả năng mở rộng (Scalability)	13
4.4 Khả năng tích hợp (Interoperability)	13
4.5 Tính khả dụng (Availability)	13
4.6 Tính dễ sử dụng (Usability)	13
4.7 Khả năng giám sát và bảo trì (Maintainability & Monitoring)	13
5 Use-case và kịch bản vận hành (Scenario)	14
5.1 Use-case toàn hệ thống	14
5.2 Sinh viên	15
5.3 Tutor	22
5.4 Khoa / Bộ môn	31
5.5 Phòng Đào tạo	32
5.6 Phòng Công tác sinh viên	34
5.7 Quản trị viên hệ thống (System Administrator)	37



5.8	Điều phối viên	41
6	Lược đồ hoạt động và trình tự (Activity and Sequence Diagram)	44
6.1	Lược đồ hoạt động	44
6.2	Lược đồ trình tự	46
7	Mockup-UI	55
7.1	Sinh viên	55
7.2	Tutor	57
7.3	Phòng Đào tạo	60
7.4	Phòng CTSV	61
7.5	Khoa - Bộ môn	62
7.6	Điều phối viên	63
8	Các lược đồ cấu trúc (Structure Diagram)	65
8.1	Lược đồ triển khai (Deployment Diagram)	65
8.2	Lược đồ hiện thực (Implementation Diagram)	66
8.2.1	Lược đồ hiện thực các component	66
8.2.2	Lược đồ hiện thực các Package	67
8.3	Lược đồ các lớp hiện thực	68



Danh sách thành viên và khối lượng công việc hoàn thành

STT	Họ và tên	MSSV	% hoàn thành
1	Nguyễn Đức Nghĩa	2312266	100%
2	Phan Huy Quang Minh	2312105	100%
3	Ngô Thanh Hùng	2311289	100%
4	Nguyễn Dắc Nghĩa	2312265	100%
5	Hà Cao Đức Minh	2312059	100%
6	Nguyễn Trí Hải	2314047	100%
7	Trần Tiến Khải	2311566	100%



1 Tổng quan về dự án

1.1 Bối cảnh và tính cấp thiết

Hiện nay, tại Trường Đại học Bách Khoa – DHQG TP.HCM, chương trình Tutor/Mentor đã được triển khai nhằm hỗ trợ sinh viên trong quá trình học tập và phát triển kỹ năng, thông qua việc kết nối họ với các giảng viên, nghiên cứu sinh hoặc sinh viên có học lực tốt. Tuy nhiên, công tác quản lý chương trình vẫn đang thực hiện chủ yếu bằng phương pháp thủ công, rườm rà và thiếu một nền tảng thống nhất, gây khó khăn trong việc theo dõi, sắp xếp lịch gấp gáp, cũng như đánh giá hiệu quả hỗ trợ. Trong bối cảnh số lượng sinh viên tham gia ngày càng tăng, nhu cầu xây dựng một hệ thống số hiện đại, đồng bộ và dễ mở rộng trở nên cấp thiết hơn bao giờ hết. Từ thực tế đó, chúng tôi đã phát triển ứng dụng BKArch với mục tiêu số hóa toàn bộ quy trình của chương trình Tutor/Mentor. Ứng dụng này sẽ giúp đơn giản hóa việc đăng ký, phân công, đặt lịch và quản lý buổi gấp, đồng thời hỗ trợ các bên liên quan dễ dàng theo dõi, đánh giá và khai thác dữ liệu để nâng cao chất lượng học tập cũng như tối ưu hóa nguồn lực của nhà trường.

1.2 Mục đích và mục tiêu dự án

1.2.1 Mục đích dự án:

Ứng dụng hỗ trợ Tutor được xây dựng nhằm:

- Cung cấp nền tảng hỗ trợ học tập và phát triển kỹ năng cho sinh viên một cách có hệ thống.
- Quản lý và vận hành chương trình Tutor/Mentor một cách hiện đại, hiệu quả và có khả năng mở rộng trong tương lai.
- Tăng cường kết nối giữa sinh viên và Mentor nhằm nâng cao chất lượng học tập và giảng dạy.
- Tích hợp chặt chẽ với hạ tầng công nghệ hiện có của nhà trường nhằm đảm bảo tính bảo mật, nhất quán dữ liệu nội bộ, đồng thời tạo nền tảng cho việc mở rộng và phát triển hệ sinh thái số hỗ trợ học tập, kết nối và quản lý toàn diện trong môi trường đại học



1.2.2 Mục tiêu dự án:

Ứng dụng được xây dựng nhằm đáp ứng các mục tiêu cụ thể sau đây:

Nhóm mục tiêu	Mô tả chi tiết
Quản lý dữ liệu người dùng	Lưu trữ thông tin cá nhân của sinh viên và tutor (chuyên môn, nhu cầu hỗ trợ, thời gian sẵn sàng). Cho phép sinh viên đăng ký chương trình và được phân công tutor (thủ công hoặc tự động).
Tổ chức, quản lý buổi tư vấn	Thiết lập, hủy hoặc thay đổi lịch hẹn. Hỗ trợ hình thức gặp trực tiếp và trực tuyến. Tích hợp thông báo tự động, nhắc lịch và ghi nhận buổi gặp (nếu cần).
Dánh giá và phản hồi	Sinh viên đánh giá chất lượng buổi tư vấn. Tutor theo dõi tiến trình học tập. Khoa/Bộ môn phân tích dữ liệu để theo dõi kết quả học tập. Phòng CTSV sử dụng dữ liệu để xét điểm rèn luyện hoặc học bổng.
Tích hợp hệ thống của trường	Kết nối hệ thống đăng nhập tập trung HCMUT_SSO. Đồng bộ dữ liệu từ HCMUT_DATACORE. Phân quyền truy cập theo vai trò. Liên kết HCMUT_LIBRARY để chia sẻ tài liệu học tập.
Tính năng nâng cao	Ghép cặp thông minh bằng AI. Cộng đồng trực tuyến cho tutor và mentee. Hỗ trợ các chương trình học thuật và phi học thuật. Cá nhân hóa hỗ trợ học tập bằng AI.

1.3 Phạm vi dự án

1.3.1 Chức năng cốt lõi

1. Quản lý người dùng & xác thực

- Đăng nhập tập trung thông qua **HCMUT_SSO**.
- Đồng bộ dữ liệu cá nhân từ **HCMUT_DATACORE**: họ tên, MSSV/Mã cán bộ, email học vụ, khoa/ngành, trạng thái học tập/giảng dạy.
- Quản lý phân quyền theo vai trò (sinh viên, tutor, điều phối viên, bộ môn, quản trị hệ thống).

2. Quản lý tutor & mentee

- Tutor quản lý hồ sơ chuyên môn, thời gian rảnh, danh sách sinh viên đang hướng dẫn.
- Sinh viên đăng ký tham gia chương trình, chọn hoặc được phân công tutor.



- Điều phối viên xem, phân công hoặc phê duyệt ghép cặp tutor sinh viên.

3. Quản lý lịch & buổi tư vấn

- Tutor tạo lịch rảnh và mở buổi tư vấn (offline hoặc online).
- Sinh viên đặt lịch gặp tutor, với hỗ trợ hủy/đổi/dời lịch.
- Gửi thông báo tự động (email, notification) khi có thay đổi lịch.
- Tự động nhắc lịch (reminder) trước buổi gặp.
- Cho phép ghi nhận nội dung và biên bản buổi gặp (nếu cần).

4. Đánh giá & phản hồi

- Sinh viên phản hồi, đánh giá chất lượng buổi tư vấn.
- Tutor theo dõi tiến bộ và ghi nhận kết quả học tập của mentee.
- Bộ môn và Phòng Đào tạo khai thác dữ liệu để tổng hợp tình hình học tập.

5. Quản lý học liệu & tài nguyên

- Tích hợp với **HCMUT_LIBRARY** để truy cập, chia sẻ tài liệu, giáo trình và sách.

1.3.2 Chức năng mở rộng (đề xuất)

1. Hệ thống AI ghép cặp và tương tác trực tuyến

- Ứng dụng AI để gợi ý phân công tutor mentee tối ưu, dựa trên hồ sơ cá nhân, chuyên môn và nhu cầu hỗ trợ.
- Tạo cộng đồng trực tuyến cho tutor mentee nhằm trao đổi, thảo luận và chia sẻ kinh nghiệm.
- Tích hợp chat nhóm/cá nhân, cho phép giao tiếp nhanh và nhận thông báo, nhắc nhở lịch hẹn ngay trong hệ thống.

2. Hỗ trợ học tập cá nhân hóa bằng AI

- AI phân tích tiến trình học tập, dữ liệu phản hồi và kết quả rèn luyện.
- Dưa ra gợi ý học tập, tài liệu hoặc phương pháp phù hợp với từng sinh viên.

3. Quản lý Seminar/Event

- Cho phép tạo và quản lý sự kiện học thuật hoặc kỹ năng mềm.
- Sinh viên đăng ký tham gia, check-in bằng QR hoặc thẻ sinh viên.
- Hệ thống quản lý số lượng tham dự và phản hồi sau sự kiện.

1.4 Định hướng kỹ thuật

1.4.1 Mô hình Server-Client

Mô hình Server-Client (Máy chủ - Máy khách) là một mô hình kiến trúc trong đó các chức năng được phân chia rõ ràng:



- **Server (Máy chủ):** Đóng vai trò là một nhà cung cấp dịch vụ (service provider). Nó là một máy tính (hoặc hệ thống phần mềm) mạnh mẽ, luôn trong trạng thái chờ đợi và lắng nghe các yêu cầu từ client. Khi nhận được yêu cầu, server sẽ xử lý và cung cấp tài nguyên, dữ liệu hoặc dịch vụ phù hợp trả về cho client.
- **Client (Máy khách):** Đóng vai trò là người yêu cầu dịch vụ (service requester). Nó là một thiết bị (máy tính, điện thoại, trình duyệt web, ứng dụng) gửi yêu cầu đến server để sử dụng tài nguyên hoặc dịch vụ mà server cung cấp.

Việc lựa chọn kiến trúc máy khách-máy chủ thay vì mạng ngang hàng (P2P) thường mang lại lợi thế khi các yêu cầu cụ thể về kiểm soát tập trung, bảo mật, khả năng mở rộng và tính tin cậy là yếu tố tối quan trọng.

Lý do nên chọn mô hình Máy khách-Máy chủ:

1. **Quản lý và kiểm soát tập trung:** Mô hình máy khách-máy chủ cho phép một cơ quan có thẩm quyền tập trung (máy chủ) quản lý tài nguyên, dữ liệu và quyền truy cập của người dùng. Điều này giúp đơn giản hóa công tác quản trị, cập nhật và bảo trì hệ thống tổng thể.
2. **Bảo mật nâng cao:** Với một máy chủ tập trung, các biện pháp bảo mật có thể được triển khai và thực thi một cách nhất quán trên tất cả các máy khách. Dữ liệu có thể được lưu trữ và bảo vệ trong một môi trường được kiểm soát chặt chẽ, đồng thời các quyền truy cập có thể được quản lý một cách chính xác. Ngược lại, các mạng P2P có thể dễ bị tổn thương hơn trước các vi phạm bảo mật vì mỗi peer (nút ngang hàng) đều là một điểm tấn công tiềm năng.
3. **Khả năng mở rộng:** Kiến trúc máy khách-máy chủ nhìn chung có khả năng mở rộng tốt hơn, cho phép bổ sung thêm nhiều máy khách và tài nguyên mà không ảnh hưởng đáng kể đến hiệu suất. Các máy chủ chuyên dụng có thể được nâng cấp và tối ưu hóa để xử lý khối lượng công việc ngày càng tăng.
4. **Độ tin cậy và khả năng sẵn sàng cao:** Một máy chủ được bảo trì tốt có thể mang lại thời gian hoạt động liên tục (uptime) và độ tin cậy cao hơn so với mạng P2P, nơi khả năng sẵn có của tài nguyên phụ thuộc vào việc từng peer riêng lẻ có đang trực tuyến và hoạt động hay không.
5. **Tính nhất quán và toàn vẹn dữ liệu:** Việc lưu trữ dữ liệu tập trung trên máy chủ đảm bảo tính nhất quán và toàn vẹn của thông tin trên tất cả các máy khách, ngăn ngừa các xung đột hoặc sai lệch tiềm ẩn có thể phát sinh trong môi trường P2P phi tập trung.
6. **Hiệu suất tốt hơn cho các ứng dụng cụ thể:** Đối với các ứng dụng đòi hỏi xử lý hiệu năng cao, lưu trữ dữ liệu lớn hoặc các thao tác phức tạp, một máy chủ chuyên dụng có thể cung cấp các tài nguyên và sức mạnh xử lý cần thiết một cách hiệu quả hơn so với các peer riêng lẻ.
7. **Khắc phục sự cố dễ dàng hơn:** Với một điểm kiểm soát tập trung, việc xác định và giải quyết các sự cố trong mạng máy khách-máy chủ thường đơn giản và trực quan hơn so với việc chẩn đoán vấn đề trên vô số các peer riêng lẻ trong một hệ thống P2P.



1.4.2 ReactJS

ReactJS là một thư viện JavaScript phổ biến dùng để xây dựng giao diện người dùng (UI) theo hướng component.

Trong project này, nhóm sử dụng ReactJS cho phần **Frontend**:

- **Quản lý UI:** Chia nhỏ giao diện thành các component tái sử dụng.
- **Tương tác động:** Kết hợp với CSS/JavaScript để tạo trải nghiệm người dùng mượt mà.
- **Render phía client:** Áp dụng mô hình *Client-Side Rendering (CSR)*.

Ưu điểm khi sử dụng ReactJS:

1. **Component-based:** Cho phép tái sử dụng và quản lý UI dễ dàng.
2. **Hiệu năng cao:** Virtual DOM giúp render nhanh và tối ưu.
3. **Hệ sinh thái phong phú:** Tích hợp dễ dàng với Redux, React Router, Next.js.
4. **Phù hợp cho SPA:** Đặc biệt thích hợp cho ứng dụng Single Page Application hiện đại.

1.4.3 ExpressJS

ExpressJS là một framework web tối giản cho Node.js, được sử dụng để xây dựng backend và API.

Trong project này, nhóm sử dụng ExpressJS cho phần **Backend**:

- **Xử lý HTTP request:** Tiếp nhận và phản hồi dữ liệu từ client.
- **Routing:** Định nghĩa các đường dẫn và API endpoint.
- **Middleware:** Quản lý xác thực, logging, xử lý lỗi.
- **Kết nối cơ sở dữ liệu:** Truy xuất và lưu trữ dữ liệu.

Ưu điểm khi sử dụng ExpressJS:

1. **Nhé và tối giản:** Cung cấp các chức năng cốt lõi, dễ học và dễ triển khai.
2. **Linh hoạt:** Cho phép tích hợp nhiều middleware và thư viện tùy chỉnh.
3. **Khả năng mở rộng:** Phù hợp cho cả ứng dụng nhỏ lẫn hệ thống lớn.
4. **Hiệu suất cao:** Tối ưu cho việc xây dựng RESTful API và microservices.

1.4.4 Git và GitHub

Git là hệ thống quản lý phiên bản phân tán, giúp ghi lại mọi thay đổi trong mã nguồn. GitHub là dịch vụ lưu trữ trên web dành cho các dự án sử dụng Git, cung cấp các công cụ để cộng tác.

Điểm cộng khi dùng Git/GitHub cho một dự án app/web:

- **Quản lý phiên bản an toàn:** Theo dõi mọi thay đổi code, dễ dàng khôi phục lại trạng thái ổn định trước đó nếu xảy ra lỗi.
- **Hợp tác nhóm hiệu quả:** Nhiều người có thể làm việc song song trên các tính năng khác nhau và hợp nhất lại một cách dễ dàng, tránh xung đột.



- **Theo dõi công việc minh bạch:** Sử dụng Issues và Pull Requests để phân công task, thảo luận và kiểm tra code trước khi đưa vào dự án chính.
- **Triển khai tự động (CI/CD):** Tích hợp với các công cụ để tự động kiểm thử, build và triển khai ứng dụng lên môi trường production.

1.5 Định hướng mở rộng và nâng cấp

Hệ thống có thể được mở rộng và tích hợp thêm các tính năng sau: tích hợp trí thông minh nhân tạo (AI), xây dựng các tính năng cộng đồng, trang diễn đàn cho sinh viên, hỗ trợ thực hiện các chương trình học thuật và phi học thuật.

1.5.1 Tích hợp AI

AI được tích hợp vào để đánh giá, phân tích tình hình học tập của sinh viên cũng như hỗ trợ sinh viên trong học tập.

- **Phân tích, đánh giá tình hình học tập của sinh viên:** Tổng hợp lại điểm số của sinh viên qua từng kỳ, đánh giá tình hình hiện tại của sinh viên, đưa ra các cảnh báo hoặc lưu ý khi thấy điểm số có biến động .
- **Hỗ trợ sinh viên trong học tập:** Dựa vào số liệu tổng hợp được và nhu cầu của từng sinh viên, đưa ra các lộ trình học tập phù hợp với từng cá nhân. Sử dụng các kỹ thuật AI để tối ưu hóa việc ghép đôi giữa giảng viên và sinh viên dựa trên các tiêu chí cụ thể như lĩnh vực chuyên môn của giảng viên và nhu cầu hỗ trợ của sinh viên.

1.5.2 Xây dựng các tính năng cộng đồng và các diễn đàn:

Cung cấp các tính năng cộng đồng và các diễn đàn trực tuyến, nơi sinh viên và giảng viên có thể kết nối, thảo luận và chia sẻ kiến thức, tạo ra một môi trường học tập và trao đổi mở rộng bên ngoài các buổi học trên lớp.

Diễn đàn thảo luận:

- **Tạo bài viết và chủ đề:** Giảng viên và sinh viên có thể đăng các câu hỏi, chia sẻ tài liệu hữu ích, và bắt đầu các cuộc thảo luận về các môn học hoặc lĩnh vực chuyên môn.
- **Phân loại chủ đề:** Chủ đề có thể được phân loại theo môn học, ngành học hoặc các thẻ (tag) liên quan để dễ dàng tìm kiếm và theo dõi.
- **Bình luận và tương tác:** Người dùng có thể bình luận, trả lời, và tương tác với bài viết của người dùng khác, trao đổi kiến thức và kinh nghiệm.

Tính năng cộng đồng:

- **Nhóm học tập:** Cho phép người dùng tạo các nhóm học tập nhỏ để cùng nhau làm bài tập hoặc ôn thi.
- **Tin nhắn riêng tư:** Các sinh viên, giảng viên có thể gửi tin nhắn cho nhau, có thể đánh dấu tin nhắn quan trọng.
- **Thông báo và cập nhật:** Thông báo cho người dùng về các tin nhắn riêng tư, các tin nhắn cộng đồng cũng như các bình luận trong bài viết mà người dùng đang theo dõi trong diễn đàn. Những tin nhắn quan trọng sẽ được làm nổi bật lên.



1.5.3 Chương trình học thuật và phi học thuật:

Hệ thống hỗ trợ các chương trình học thuật và phi học thuật do nhà trường hoặc bên thứ ba thông qua nhà trường tổ chức.

- **Tổ chức chương trình:** Những chương trình tổ chức sẽ được tổng hợp lại trong một trang riêng. Phân loại ra các hoạt động đã tổ chức và đang tổ chức. Mỗi hoạt động sẽ có thông tin chi tiết và cách tham dự.
- **Hỗ trợ sinh viên:** Hệ thống sẽ lưu trữ lại và hiển thị ra những chương trình mà sinh viên đã tham dự.

1.5.4 Cá nhân hóa học tập:

Hệ thống áp dụng AI để đưa ra phương pháp học tập, tài liệu học tập phù hợp với nhu cầu và mục tiêu của mỗi sinh viên.

- **Phân tích tình hình học tập:** Theo dõi tiến độ học tập và xác định các môn sinh viên đang gặp khó khăn.
- **Đề xuất phương pháp học tập phù hợp:** Đề xuất các tài liệu, bài giảng, bài tập hoặc các buổi tư vấn chuyên sâu phù hợp với điểm yếu của từng sinh viên. Sinh viên có thể lựa chọn phương pháp học tập phù hợp như tự học, học nhóm hoặc tư vấn 1-1 với giảng viên.



2 Các bên liên quan (Relevant Stakeholders)

Nhóm Stakeholder	Vai trò và yêu cầu
Sinh viên (Mentee)	Đăng ký tham gia chương trình, lựa chọn hoặc được gợi ý tutor phù hợp, đặt lịch tư vấn, phản hồi chất lượng buổi học, nhận hỗ trợ học tập cá nhân hóa.
Tutor (Giảng viên, Nghiên cứu sinh, Sinh viên năm trên)	Thiết lập lịch rảnh, mở buổi tư vấn, quản lý buổi gặp, ghi nhận tiến bộ mentee, chia sẻ tài liệu học tập, nhận phản hồi từ mentee.
Khoa / Bộ môn	Khai thác dữ liệu đánh giá để nắm tình hình học tập của sinh viên, hỗ trợ phân bổ nguồn lực giảng dạy.
Phòng Đào tạo (PDT)	Sử dụng báo cáo tổng quan để tối ưu hóa phân bổ nguồn lực, theo dõi hiệu quả chương trình.
Phòng Công tác Sinh viên (P.CTSV)	Căn cứ vào kết quả tham gia để cộng điểm rèn luyện, xét học bổng, hỗ trợ phát triển kỹ năng mềm.
Quản trị viên hệ thống	Thiết kế, xây dựng, bảo trì hệ thống, tích hợp các dịch vụ nội bộ, triển khai các tính năng nâng cao như AI ghép cặp, công đồng trực tuyến.
Điều phối viên	Quản lý vận hành chương trình, phân bổ tutor, theo dõi tiến độ, xử lý các vấn đề phát sinh, tổng hợp báo cáo.
Trung tâm xác thực (HC-MUT_SSO)	Cung cấp dịch vụ xác thực tập trung.
Trung tâm dữ liệu (HC-MUT_DATACORE)	Đồng bộ dữ liệu cá nhân.
Thư viện HCMUT (HC-MUT_LIBRARY)	Cung cấp tài nguyên học tập, hỗ trợ chia sẻ tài liệu giữa tutor và mentee.

3 Yêu cầu chức năng (Functional Requirements)

3.1 Quản lý người dùng & xác thực

Hệ thống cho phép đăng nhập qua HCMUT_SSO với thời gian xử lý trung bình dưới 2 giây, đảm bảo thao tác nhanh và dễ dàng. Dữ liệu được đồng bộ từ HCMUT_DATACORE với tỷ lệ thành công đạt trên 99%, hiển thị chính xác và đầy đủ. Cơ chế phân quyền theo vai trò hoạt động chính xác trên 99%, giúp người dùng truy cập đúng quyền hạn.

3.2 Quản lý tutor & mentee

Hệ thống hỗ trợ quản lý hồ sơ với tỷ lệ cập nhật thành công trên 98%, đảm bảo tutor thao tác thuận tiện. Quá trình đăng ký và ghép cặp có tỷ lệ thành công trên 95%, mức độ phù hợp trung



bình đạt từ 4/5 điểm trở lên. Việc phê duyệt ghép cặp được xử lý trong vòng 5 giây, giao diện được đánh giá dễ sử dụng bởi điều phối viên.

3.3 Quản lý lịch & buổi tư vấn

Sinh viên có thể đặt lịch tư vấn với tỷ lệ thành công trên 98%, mức độ hài lòng đạt từ 4/5 điểm. Hệ thống gửi thông báo và nhắc lịch đúng giờ với độ chính xác trên 99%, tutor xác nhận hiệu quả. Nội dung buổi gặp được ghi nhận với tỷ lệ lưu biên bản trên 95%, đảm bảo rõ ràng và dễ truy xuất.

3.4 Dánh giá & phản hồi

Sau buổi học, sinh viên phản hồi với tỷ lệ trên 80%, mức độ đánh giá trung bình đạt từ 4/5 điểm. Tutor có thể theo dõi tiến bộ của mentee với tỷ lệ ghi nhận trên 95%, công cụ trực quan và dễ sử dụng. Dữ liệu học tập được tổng hợp đầy đủ trên 90%, báo cáo được bộ môn đánh giá hữu ích.

3.5 Quản lý học liệu & tài nguyên

Sinh viên truy cập tài liệu từ HCMUT_LIBRARY với tỷ lệ thành công trên 99%, đánh giá dễ tìm kiếm. Hệ thống hỗ trợ chia sẻ học liệu với tần suất trung bình trên 3 lần/người/tháng, tutor xác nhận tích hợp thuận tiện.

3.6 Chức năng mở rộng (đề xuất)

AI ghép cặp & tương tác trực tuyến

AI hỗ trợ ghép cặp tối ưu với tỷ lệ trùng khớp trên 85%, người dùng đánh giá mức độ phù hợp từ 4/5 điểm. Hệ thống chat trực tuyến có số lượt tương tác trung bình trên 5 lần/người/tháng, mức độ hài lòng đạt trên 80%.

Hỗ trợ học tập cá nhân hóa bằng AI

AI đưa ra gợi ý học tập cá nhân hóa cho trên 90% sinh viên, được đánh giá hữu ích từ 4/5 điểm. Kết quả học tập cải thiện trên 70%, tutor xác nhận AI hỗ trợ đúng trọng tâm.

Quản lý Seminar/Event

Hệ thống hỗ trợ check-in sự kiện với tỷ lệ thành công trên 98%, sinh viên đánh giá sự kiện hữu ích. Sau sự kiện, tỷ lệ phản hồi đạt trên 75%, ban tổ chức xác nhận hệ thống hiệu quả.

4 Yêu cầu phi chức năng (Non-Functional Requirements)

4.1 Hiệu năng (Performance)

Hệ thống phải xử lý yêu cầu đăng ký, đặt lịch và truy xuất thông tin trong vòng dưới 2 giây. Hệ thống có khả năng phục vụ đồng thời tối thiểu 5000 người dùng mà không bị gián đoạn. Thời gian phản hồi trung bình cho các thao tác thường xuyên không vượt quá 1 giây.

4.2 Bảo mật (Security)

Tích hợp xác thực tập trung HCMUT_SSO để đảm bảo đăng nhập an toàn và thống nhất. Dữ liệu cá nhân phải được mã hóa khi lưu trữ và truyền tải. Phân quyền truy cập theo vai trò được đồng bộ từ hệ thống tập trung. Ghi nhận nhật ký truy cập và thao tác để phục vụ kiểm tra bảo mật.

4.3 Khả năng mở rộng (Scalability)

Hệ thống phải có khả năng mở rộng để phục vụ số lượng người dùng tăng theo từng học kỳ. Kiến trúc hệ thống phải hỗ trợ bổ sung các chức năng nâng cao như AI ghép cặp hoặc cộng đồng trực tuyến.

4.4 Khả năng tích hợp (Interoperability)

Tích hợp với HCMUT_DATACORE để đồng bộ dữ liệu cá nhân và trạng thái học tập. Kết nối với HCMUT_LIBRARY để chia sẻ tài liệu học tập. Hỗ trợ API để tích hợp với các hệ thống nội bộ khác.

4.5 Tính khả dụng (Availability)

Hệ thống phải hoạt động liên tục 24/7 với thời gian uptime tối thiểu 99.5% mỗi tháng. Có cơ chế sao lưu và phục hồi dữ liệu định kỳ.

4.6 Tính dễ sử dụng (Usability)

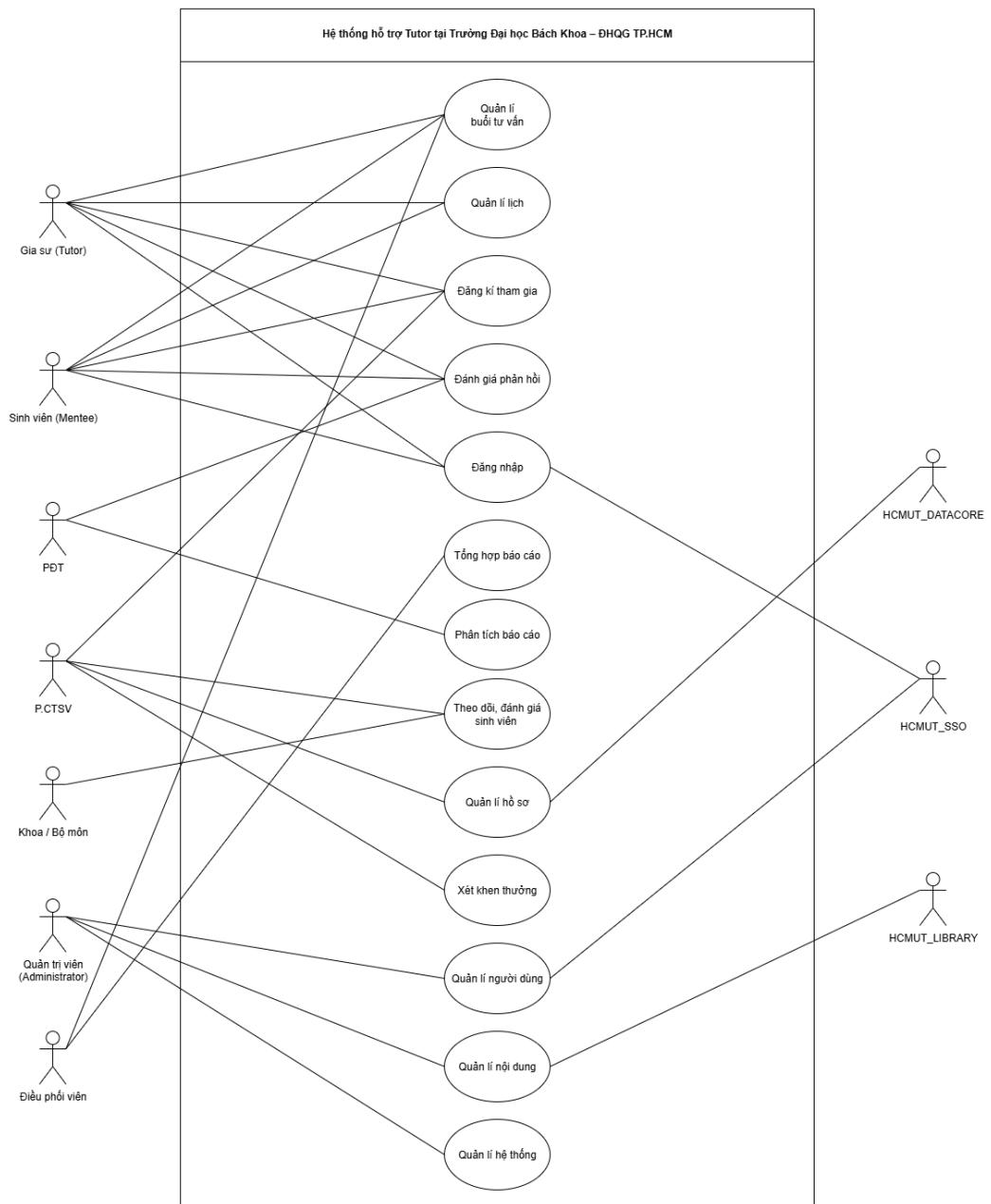
Giao diện thân thiện, hỗ trợ đa thiết bị (máy tính, điện thoại). Có hướng dẫn sử dụng rõ ràng cho từng nhóm người dùng. Hệ thống hỗ trợ tiếng Việt và tiếng Anh.

4.7 Khả năng giám sát và bảo trì (Maintainability & Monitoring)

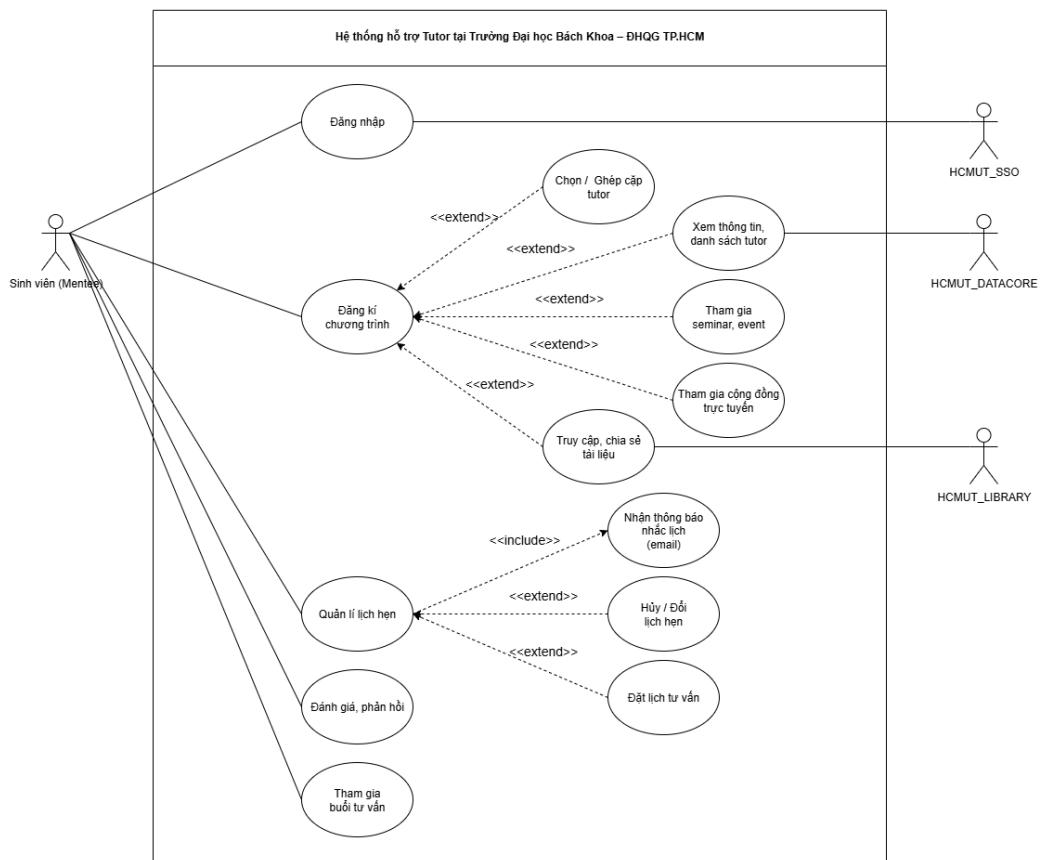
Cho phép theo dõi tiến độ học tập, phản hồi chất lượng buổi học và tổng hợp báo cáo cho các phòng ban. Hệ thống dễ bảo trì, cập nhật và nâng cấp.

5 Use-case và kịch bản vận hành (Scenario)

5.1 Use-case toàn hệ thống



5.2 Sinh viên





Use-case ID	UC-STU-001
Use-case	Dăng nhập HCMUT_SSO
Actor	Sinh viên, HCMUT_SSO
Description	Cho phép sinh viên đăng nhập vào hệ thống thông qua HCMUT_SSO. Nếu là lần đầu đăng nhập, hệ thống sẽ tự động đồng bộ hồ sơ cơ bản từ HCMUT_DATACORE để đảm bảo dữ liệu đầy đủ.
Precondition	1. Dịch vụ HCMUT_SSO đang hoạt động ổn định. 2. Sinh viên chưa có phiên đăng nhập hợp lệ trên thiết bị.
Postcondition	Phiên đăng nhập hợp lệ được tạo; hồ sơ cơ bản của sinh viên được đồng bộ vào hệ thống.
Trigger	Sinh viên chọn nút “Đăng nhập bằng HCMUT_SSO”.
Normal Flow	1. Hệ thống chuyển hướng đến trang đăng nhập SSO. 2. Sinh viên nhập tên tài khoản và mật khẩu. 3. Hệ thống SSO xác thực thông tin đăng nhập. 4. Nếu hợp lệ, hệ thống tạo phiên đăng nhập, gán vai trò <i>student</i> . 5. Nếu là lần đầu, hệ thống đồng bộ hồ sơ từ DATACORE.
Alternative Flow	Không có.
Exception Flow	E1. Sai thông tin đăng nhập hoặc hết thời gian chờ: hệ thống hiển thị thông báo lỗi và cho phép thử lại. E2. Mất kết nối với dịch vụ SSO: hệ thống báo lỗi kết nối và yêu cầu thử lại sau.



Use-case ID	UC-STU-002
Use-case	Dăng ký chương trình Tutor/Mentor
Actor	Sinh viên
Description	Sinh viên đăng ký tham gia chương trình hỗ trợ học tập. Có thể chọn tutor cụ thể từ danh sách hoặc yêu cầu hệ thống tự động ghép cặp dựa trên hồ sơ và lịch rảnh.
Precondition	Sinh viên đã đăng nhập hợp lệ vào hệ thống.
Postcondition	Liên kết mentee-tutor được tạo hoặc yêu cầu ghép cặp được ghi nhận và chờ duyệt.
Trigger	Sinh viên mở mục “Đăng ký chương trình”.
Normal Flow	<ol style="list-style-type: none">1. Hệ thống hiển thị form đăng ký chương trình.2. Sinh viên chọn một trong hai cách:<ol style="list-style-type: none">a. Chọn tutor cụ thể từ danh sách.b. Yêu cầu hệ thống ghép cặp tự động.3. Hệ thống ghi nhận yêu cầu và trả về kết quả ghép cặp hoặc trạng thái chờ duyệt.
Alternative Flow	Không có.
Exception Flow	<p>E1. Không có tutor phù hợp: hệ thống gợi ý 3 lựa chọn gần nhất để sinh viên chọn thủ công.</p> <p>E2. Lỗi kết nối hoặc ghi nhận yêu cầu thất bại: hệ thống hiển thị thông báo “Không thể đăng ký, vui lòng thử lại”.</p>



Use-case ID	UC-STU-003
Use-case	Xem thông tin, danh sách tutor
Actor	Sinh viên
Description	Cho phép sinh viên tra cứu hồ sơ, chuyên môn và lịch rảnh của tutor để hỗ trợ việc đăng ký hoặc ghép cặp.
Precondition	Sinh viên đã đăng nhập hợp lệ vào hệ thống.
Postcondition	Không thay đổi dữ liệu; tùy chọn lưu bộ lọc tìm kiếm gần nhất để sử dụng lại.
Trigger	Sinh viên chọn mục “Danh sách tutor” hoặc truy cập từ UC đăng ký chương trình.
Normal Flow	<ol style="list-style-type: none">Hệ thống hiển thị danh sách tutor kèm chức năng tìm kiếm và lọc.Sinh viên chọn một tutor để xem chi tiết hồ sơ.Sinh viên xem lịch rảnh của tutor và quay lại màn hình đăng ký.
Alternative Flow	Không có.
Exception Flow	E1. Lỗi tải danh sách tutor: hệ thống hiển thị thông báo lỗi và nút “Thử lại”. E2. Kết nối mạng không ổn định: danh sách hiển thị không đầy đủ, hệ thống cảnh báo và yêu cầu tải lại.

Use-case ID	UC-STU-004
Use-case	Đặt lịch tư vấn
Actor	Sinh viên
Description	Cho phép sinh viên đặt lịch tư vấn với tutor (offline hoặc online). Hệ thống sẽ gửi thông báo và thiết lập nhắc lịch tự động.
Precondition	Sinh viên đã có liên kết mentee-tutor hợp lệ.
Postcondition	Lịch hẹn được lưu vào hệ thống; reminder được thiết lập và gửi thông báo cho các bên liên quan.
Trigger	Sinh viên chọn chức năng “Đặt lịch”.
Normal Flow	<ol style="list-style-type: none">Hệ thống hiển thị lịch rảnh của tutor.Sinh viên chọn slot hợp lệ và xác nhận đặt lịch.Hệ thống lưu lịch hẹn, gửi thông báo cho tutor và thiết lập reminder.
Alternative Flow	Không có.
Exception Flow	E1. Slot vừa bị chiếm hoặc trùng: hệ thống hiển thị cảnh báo và đề xuất slot khác. E2. Lỗi lưu lịch do sự cố hệ thống: hiển thị thông báo “Không thể lưu, vui lòng thử lại”.



Use-case ID	UC-STU-005
Use-case	Hủy/Dổi lịch tư vấn
Actor	Sinh viên
Description	Cho phép sinh viên hủy hoặc đổi lịch tư vấn sang slot khác. Hệ thống sẽ cập nhật lịch và gửi thông báo cho các bên liên quan.
Precondition	Lịch hẹn đang ở trạng thái <i>Scheduled</i> .
Postcondition	Lịch hẹn phản ánh trạng thái mới; reminder được cập nhật hoặc hủy bỏ.
Trigger	Sinh viên mở chi tiết lịch hẹn.
Normal Flow	1. Sinh viên chọn “Hủy” hoặc “Đổi lịch”. 2. Nếu đổi: sinh viên chọn slot mới. 3. Hệ thống cập nhật lịch, gửi thông báo cho tutor và điều chỉnh reminder.
Alternative Flow	Không có.
Exception Flow	E1. Slot mới không hợp lệ hoặc đã bị chiếm: hệ thống yêu cầu chọn lại. E2. Lỗi cập nhật lịch do sự cố hệ thống: hiển thị thông báo “Không thể cập nhật, vui lòng thử lại”.

Use-case ID	UC-STU-006
Use-case	Tham gia buổi tư vấn
Actor	Sinh viên
Description	Cho phép sinh viên tham gia buổi tư vấn theo lịch đã đặt, có thể là trực tiếp hoặc trực tuyến.
Precondition	1. Sinh viên có lịch hẹn hợp lệ. 2. Sinh viên đã nhận thông báo nhắc lịch (reminder).
Postcondition	Buổi tư vấn được đánh dấu trạng thái <i>Completed</i> hoặc <i>No-show</i> .
Trigger	Đến giờ hẹn hoặc sinh viên chọn “Tham gia” (đối với buổi trực tuyến).
Normal Flow	1. Sinh viên tham gia đúng kênh/địa điểm theo lịch. 2. (Tuỳ chọn) Sinh viên ghi chú hoặc lưu bản buổi tư vấn. 3. Kết thúc buổi, hệ thống đánh dấu trạng thái <i>Completed</i> .
Alternative Flow	Không có.
Exception Flow	E1. Một bên vắng mặt: hệ thống đánh dấu trạng thái <i>No-show</i> và gửi thông báo cho các bên liên quan. E2. Lỗi kết nối đối với buổi trực tuyến: hệ thống hiển thị cảnh báo và cho phép thử lại.



Use-case ID	UC-STU-008
Use-case	Truy cập & chia sẻ tài liệu (HCMUT_LIBRARY)
Actor	Sinh viên, HCMUT_LIBRARY
Description	Cho phép sinh viên tra cứu, tải xuống và (nếu được phép) chia sẻ liên kết tài liệu học tập thông qua tích hợp với HCMUT_LIBRARY.
Precondition	Sinh viên đã đăng nhập hợp lệ và có quyền truy cập tài liệu.
Postcondition	Nhật ký truy cập và chia sẻ tài liệu được ghi nhận vào hệ thống.
Trigger	Sinh viên mở mục “Học liệu”.
Normal Flow	1. Sinh viên tìm kiếm hoặc duyệt tài liệu theo bộ lọc. 2. Xem chi tiết tài liệu. 3. Tải xuống hoặc sao chép liên kết chia sẻ.
Alternative Flow	Không có.
Exception Flow	E1. Sinh viên hết quyền hoặc thiếu scope truy cập: hệ thống hiển thị thông báo “Truy cập bị từ chối”. E2. Lỗi kết nối với HCMUT_LIBRARY: hệ thống hiển thị cảnh báo và yêu cầu thử lại.

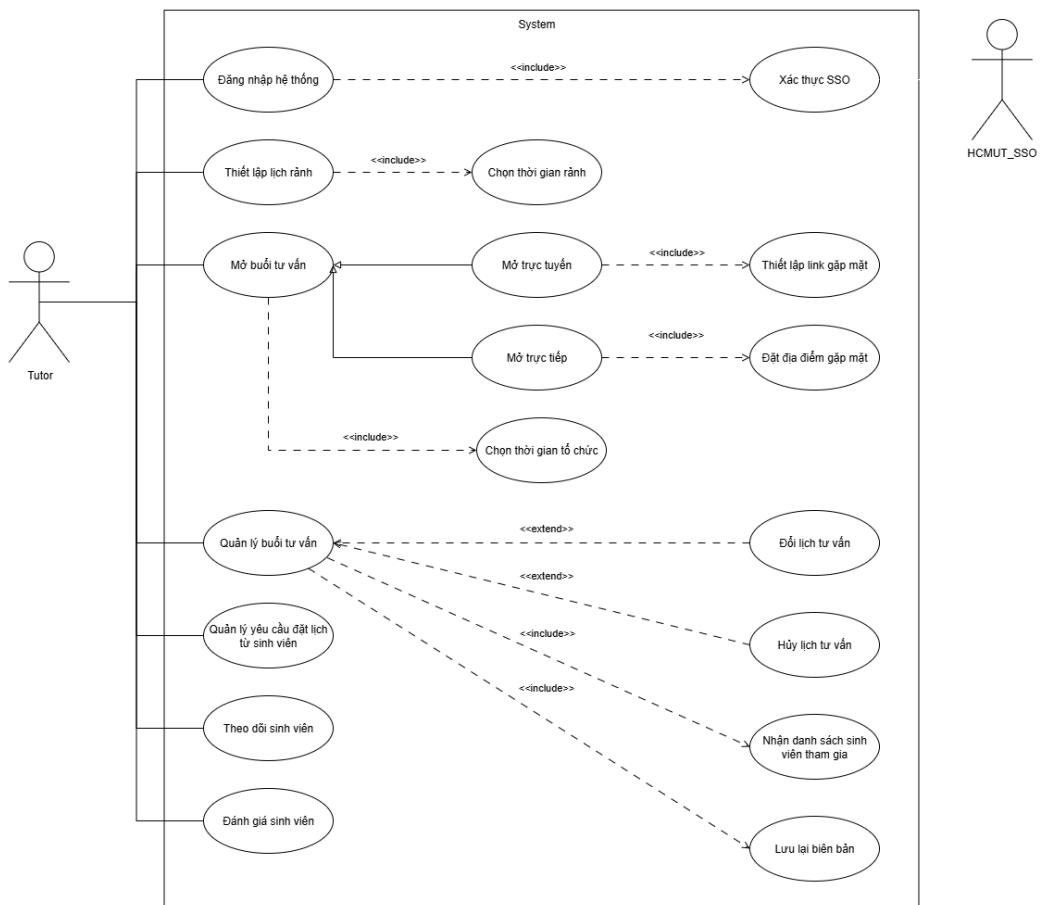
Use-case ID	UC-STU-008
Use-case	Truy cập & chia sẻ tài liệu (HCMUT_LIBRARY)
Actor	Sinh viên, HCMUT_LIBRARY
Description	Cho phép sinh viên tra cứu, tải xuống và (nếu được phép) chia sẻ liên kết tài liệu học tập thông qua tích hợp với HCMUT_LIBRARY.
Precondition	Sinh viên đã đăng nhập hợp lệ và có quyền truy cập tài liệu.
Postcondition	Nhật ký truy cập và chia sẻ tài liệu được ghi nhận vào hệ thống.
Trigger	Sinh viên mở mục “Học liệu”.
Normal Flow	1. Sinh viên tìm kiếm hoặc duyệt tài liệu theo bộ lọc. 2. Xem chi tiết tài liệu. 3. Tải xuống hoặc sao chép liên kết chia sẻ.
Alternative Flow	Không có.
Exception Flow	E1. Sinh viên hết quyền hoặc thiếu scope truy cập: hệ thống hiển thị thông báo “Truy cập bị từ chối”. E2. Lỗi kết nối với HCMUT_LIBRARY: hệ thống hiển thị cảnh báo và yêu cầu thử lại.



Use-case ID	UC-STU-009
Use-case	Tham gia seminar/event
Actor	Sinh viên
Description	Cho phép sinh viên đăng ký, nhận nhắc lịch, tham dự và gửi phản hồi sau các sự kiện học thuật hoặc kỹ năng.
Precondition	Sinh viên đã đăng nhập hợp lệ.
Postcondition	Trạng thái đăng ký, tham dự và phản hồi được cập nhật vào hệ thống.
Trigger	Sinh viên mở mục “Seminar/Event”.
Normal Flow	1. Sinh viên chọn sự kiện và xem mô tả, thời gian, hình thức. 2. Đăng ký tham dự sự kiện. 3. Nhận nhắc lịch trước giờ diễn ra và tham dự. 4. Gửi phản hồi sau sự kiện.
Alternative Flow	A1. Sinh viên hủy đăng ký trước giờ diễn ra: hệ thống giải phóng suất và cập nhật trạng thái.
Exception Flow	E1. Sự kiện hết chỗ hoặc đã đóng đăng ký: hệ thống hiển thị thông báo và gợi ý sự kiện khác. E2. Lỗi ghi nhận đăng ký: hệ thống hiển thị thông báo “Không thể đăng ký, vui lòng thử lại”.

Use-case ID	UC-STU-010
Use-case	Tham gia công đồng trực tuyến
Actor	Sinh viên
Description	Cho phép sinh viên truy cập các kênh/nhóm trao đổi với tutor và mentee khác, đồng thời nhận thông báo trong hệ thống.
Precondition	Sinh viên đã đăng nhập hợp lệ vào hệ thống.
Postcondition	Tin nhắn và trao đổi được ghi nhận theo từng kênh; thông báo được gửi theo cấu hình cá nhân.
Trigger	Sinh viên mở mục “Công đồng”.
Normal Flow	1. Hệ thống hiển thị danh sách các kênh/nhóm công đồng. 2. Sinh viên chọn một kênh để tham gia. 3. Sinh viên gửi và nhận tin nhắn trong kênh. 4. (Tuỳ chọn) Sinh viên bật hoặc tắt thông báo theo từng kênh.
Alternative Flow	A1. Sinh viên rời kênh hoặc tắt thông báo khi không còn nhu cầu tham gia.
Exception Flow	E1. Sinh viên không đủ quyền truy cập kênh: hệ thống hiển thị thông báo “Permission denied”. E2. Lỗi tải dữ liệu kênh: hệ thống hiển thị cảnh báo và cho phép thử lại.

5.3 Tutor





Use-case ID	UC-TUTOR-001
Use-case	Dăng nhập vào hệ thống
Actor	Tutor
Description	Cho phép Tutor đăng nhập xác thực thông qua HCMUT_SSO
Precondition	<ol style="list-style-type: none">1. Tutor đã có hệ thống trên HCMUT_SSO và được gán quyền Tutor trong hệ thống.2. Kết nối mạng của SSO và hệ thống Tutor đều đang trực tuyến và ổn định.3. Thông tin của Tutor đã được đồng bộ từ DATACORE.4. Hiện tại đang không có phiên đăng nhập nào trên thiết bị của Tutor.
Trigger	Tutor bấm vào nút "Đăng nhập"
Normal Flow	<ol style="list-style-type: none">1. Hệ thống sẽ kiểm tra trạng thái phiên, nếu không có phiên hiện tại sẽ chuyển đến trang đăng nhập xác thực HCMUT_SSO kèm thông tin client và URL quay lại.2. Hệ thống SSO hiện form đăng nhập.3. Tutor tiến hành nhập tên tài khoản và mật khẩu.4. Tutor ấn vào "Đăng nhập".5. Hệ thống SSO bắt đầu quá trình xác thực thông tin đầu vào của Tutor.6. Nếu thông tin hợp lệ, hệ thống SSO sẽ xác nhận đăng nhập thành công, trả về dashboard của hệ thống Tutor.
Alternative Flow	Không có
Postcondition	Tutor được xác thực và hệ thống Tutor hiển thị ra dashboard của người dùng đã được đăng nhập
Exception Flow	<p>E1. Mất kết nối với SSO: Hệ thống Tutor sẽ báo lỗi mất kết nối tới SSO.</p> <p>E2. Sai thông tin đăng nhập: Ở bước 5, nếu hệ thống xác thực thông tin nhận vào không hợp lệ sẽ hiển thị lỗi theo hệ thống SSO.</p>



Use-case ID	UC-TUTOR-002
Use-case	Thiết lập lịch rảnh
Actor	Tutor
Description	Cho phép Tutor định nghĩa các khung thời gian có thể tiếp nhận buổi hỗ trợ, đảm bảo hệ thống có dữ liệu slot chính xác để sinh viên đặt lịch.
Precondition	<ol style="list-style-type: none">Tutor đã đăng nhập thành công qua HCMUT_SSO và phiên vẫn còn hiệu lực.Hồ sơ Tutor đã được đồng bộ từ HCMUT_DATACORE, trạng thái “đang hoạt động”.Hệ thống đã tải cấu hình lịch hiện tại và các ràng buộc (thời lượng tối thiểu, khung giờ được phép).
Trigger	Tutor chọn chức năng “Thiết lập lịch rảnh” từ dashboard hoặc nhận nhắc từ hệ thống yêu cầu cập nhật lịch.
Normal Flow	<ol style="list-style-type: none">Tutor mở màn hình “Lịch rảnh”.Hệ thống hiển thị lịch tuần hiện tại cùng các slot đã thiết lập (nếu có).Tutor chọn ngày và khung giờ muốn mở slot.Hệ thống kiểm tra xung đột với các buổi đã xác nhận hoặc slot rảnh đã tồn tại.Tutor xác nhận tạo slot; hệ thống ghi nhận vào cơ sở dữ liệu.Hệ thống cập nhật trạng thái lịch, đồng bộ tới module đặt lịch và hàng đợi thông báo.Tutor lặp lại cho các khung giờ khác hoặc chỉnh sửa slot bằng thao tác kéo/thay đổi thời lượng.Tutor lưu thay đổi tổng thể; hệ thống hiển thị thông báo “Cập nhật lịch thành công”.
Alternative Flow	<ol style="list-style-type: none">Hủy slot đã đăng ký vì lý do phù hợp.<ol style="list-style-type: none">Tại bước 2, Tutor chọn slot đã công bố nhưng không còn phù hợp.Tutor chuyển trạng thái slot sang “Unavailable tạm thời” kèm lý do.Hệ thống ghi nhận thay đổi, gửi thông báo tới sinh viên đã đặt (nếu có) và quay lại bước 7 để lưu tổng thể.
Postcondition	<ol style="list-style-type: none">Các slot rảnh mới/điều chỉnh được lưu vào hệ thống, đánh dấu trạng thái “Available” và gắn với Tutor tương ứng.Lịch hiển thị cho sinh viên được cập nhật trong vòng 5 phút.Màn hình thời khóa biểu hiện thị slot thời gian vừa thêm vào.
Exception Flow	<ol style="list-style-type: none">Slot trùng với buổi đã xác nhận: Bước 4 phát hiện xung đột; hệ thống hiển thị lỗi “Khung giờ đã có buổi tutor khác”, không cho lưu slot đó, quay về bước 3.Lưu lịch thất bại do sự cố hệ thống: Tại bước 5/6 gặp lỗi kết nối; hệ thống thông báo “Không thể lưu, vui lòng thử lại”, ghi log lỗi và giữ lại dữ liệu đã nhập để Tutor thao tác lại sau.



Use-case ID	UC-TUTOR-003
Use-case	Mở buổi tư vấn
Actor	Tutor
Description	Cho phép Tutor tạo buổi tư vấn (trực tiếp hoặc trực tuyến) với thông tin thời gian, hình thức, địa điểm/link nhằm phục vụ sinh viên đã đăng ký hoặc chờ ghép.
Precondition	1. Tutor đã đăng nhập hợp lệ vào hệ thống qua HCMUT_SSO. 2. Lịch rảnh của Tutor đã được thiết lập với các slot khả dụng (UC "Thiết lập lịch rảnh"). 3. Tại slot này đã có đủ số lượng sinh viên đăng ký tối thiểu là 30 sinh viên.
Trigger	Tại dashboard, Tutor ấn vào nút "Mở buổi tư vấn"
Normal Flow	1. Tutor mở trang "Mở buổi tư vấn". 2. Hệ thống hiển thị các slot rảnh đã đủ điều kiện mở buổi tư vấn. 3. Tutor chọn hình thức buổi: "Trực tiếp" hoặc "Trực tuyến". 4a. Nếu "Trực tiếp": Tutor sẽ không cần nhập mục này, hệ thống sẽ chọn phòng phù hợp với số lượng sinh viên. 4b. Nếu "Trực tuyến": Tutor nhập/dính kèm URL phòng họp (Teams/Zoom) và mật khẩu nếu có. 5. Tutor nhập tiêu đề buổi, chủ đề chính. 6. Tutor chọn phạm vi mentee (tất cả nhóm, từng mentee, hoặc mở rộng khai cho mentee đủ điều kiện). 7. Hệ thống kiểm tra xung đột lịch. 8. Tutor xác nhận; hệ thống tạo bản ghi buổi tư vấn, cập nhật lịch và đưa vào hàng đợi thông báo. 9. Hệ thống hiển thị thông báo "Mở buổi tư vấn thành công" kèm thông tin tóm tắt và tùy chọn chỉnh sửa.
Alternative Flow	Không có
Postcondition	1. Buổi tư vấn mới được ghi nhận trạng thái "Open" gắn với slot tương ứng. 2. Thông báo/nhắc lịch được gửi tới mentee. 3. Lịch Tutor cập nhật, buổi sẵn sàng cho mentee đặt/confirm.
Exception Flow	E1. Thiếu thông tin địa điểm/URL: Bước 4b bỏ trống trường bắt buộc; hệ thống cảnh báo và yêu cầu nhập trước khi tiếp tục. E2. URL trực tuyến không hợp lệ: Hệ thống kiểm tra định dạng URL; nếu sai, hiển thị lỗi và cho thêm mới. E3. Lỗi mở buổi thất bại do sự cố hệ thống: Hệ thống trả về "Lỗi hệ thống"

Use-case ID	UC-TUTOR-004
Use-case	Quản lý buổi tư vấn



Actor	Tutor
Description	Cho phép Tutor theo dõi và điều chỉnh vòng đời của các buổi tư vấn đã mở (đổi lịch, hủy, cập nhật danh sách tham gia, ghi biên bản) nhằm bảo đảm buổi diễn ra đúng kế hoạch và dữ liệu được cập nhật nhất quán.
Precondition	1. Tutor đã đăng nhập hợp lệ (qua HCMUT_SSO). 2. Tồn tại ít nhất một buổi tư vấn ở trạng thái “Open” hoặc “Scheduled” do Tutor phụ trách. 3. Dữ liệu mentee tham gia buổi đã được hệ thống ghi nhận (từ yêu cầu đặt lịch hoặc danh sách được điều phối viên gán).
Trigger	Tutor từ dashboard chọn “Quản lý buổi tư vấn”.
Normal Flow	1. Tutor mở trang danh sách buổi tư vấn của mình. 2. Hệ thống hiển thị các buổi ở các trạng thái (Scheduled, Pending feedback, Completed) kèm bộ lọc thời gian. 3. Tutor chọn một buổi cần quản lý và xem chi tiết (thời gian, hình thức, mentee đăng ký, lịch sử thay đổi). 4. Tutor cập nhật thông tin buổi: điều chỉnh thời gian/địa điểm/link nếu cần, hoặc xác nhận buổi sẽ diễn ra như kế hoạch. 5. Hệ thống kiểm tra xung đột và quyền chỉnh sửa; nếu hợp lệ, tạm giữ slot mới và cập nhật booking liên quan. 6. Tutor (nếu buổi đã hoàn thành) nhập biên bản, ghi lại nội dung hỗ trợ, tài liệu đính kèm, kết quả đạt được. 7. Tutor lưu thay đổi; hệ thống ghi nhận cập nhật, phát thông báo tới mentee/diều phối viên và yêu cầu mentee xác nhận (nếu có). 8. Trang chi tiết hiển thị trạng thái mới cùng thông điệp thành công; Tutor có thể quay lại danh sách hoặc thao tác buổi khác.
Alternative Flow	A1 - Đổi lịch tư vấn: a. Ở bước 4, Tutor chọn “Đổi lịch” → nhập ngày/giờ mới. b. Hệ thống đề xuất slot trống gần nhất; nếu Tutor chấp nhận, quay lại bước 5 và 7 để cập nhật. A2 - Hủy lịch tư vấn: a. Bước 4 Tutor chọn “Hủy buổi” và nhập lý do. b. Hệ thống cập nhật trạng thái buổi = “Cancelled”, gửi thông báo hủy cho mentee và giải phóng slot khỏi lịch. A3 - Nhận danh sách sinh viên tham gia: a. Tutor tại bước 3 chọn tùy chọn “Xuất danh sách” → hệ thống tạo danh sách mentee đã confirm. b. Tutor dùng danh sách để chuẩn bị tài liệu; luồng trở lại bước 3. A4 - Lưu biên bản sau buổi: a. Sau bước 6, Tutor lưu biên bản và đánh dấu buổi “Completed”. b. Hệ thống gửi yêu cầu mentee phản hồi và cập nhật tiến độ học tập.



Postcondition	1. Trạng thái và thông tin buổi tư vấn được cập nhật (đổi lịch/hủy/ghi chú) và đồng bộ đến lịch của mentee liên quan. 2. Nếu buổi bị hủy hoặc đổi, thông báo được phát tới mentee và điều phối viên; biên bản được lưu sau khi Tutor hoàn tất.
Exception Flow	E1 - Có gắng đổi lịch nhưng slot mới trùng: Bước 5 phát hiện xung đột với buổi khác; hệ thống hiển thị lỗi và yêu cầu Tutor chọn slot khác (không cập nhật). E2 - Hủy buổi sau hạn cho phép: Bước 4 phát hiện buổi sắp diễn ra dưới ngưỡng hủy (≤ 2 giờ). E3 - Lưu biên bản thất bại: Bước 6 gặp lỗi lưu trữ; hệ thống giữ bản nháp biên bản, báo Tutor thử lại và không thay đổi trạng thái buổi. E4 - Tutor thiếu quyền chỉnh sửa: Bước 5 phát hiện buổi do điều phối viên khóa; hệ thống từ chối thao tác, hiển thị thông điệp “Buổi tạm khóa, liên hệ điều phối viên”.



Use-case ID	UC-TUTOR-005
Use-case	Quản lý yêu cầu đặt lịch từ sinh viên
Actor	Tutor
Description	Tutor xử lý các yêu cầu đặt lịch mà sinh viên gửi vào slot rảnh: xem chi tiết, chấp thuận, từ chối hoặc đề xuất thời gian khác nhằm đảm bảo buổi tư vấn được sắp xếp hiệu quả.
Precondition	1. Tutor đã đăng nhập hợp lệ qua HCMUT_SSO. 2. Có slot rảnh do Tutor thiết lập và sinh viên đã gửi yêu cầu đặt lịch tương ứng. 3. Thông tin sinh viên, trạng thái ghép cặp được đồng bộ từ HCMUT_DATACORE.
Trigger	Tutor mở mục “Yêu cầu đặt lịch” từ dashboard hoặc bấm vào thông báo có yêu cầu mới cần duyệt.
Normal Flow	1. Tutor truy cập danh sách yêu cầu đặt lịch đang chờ xử lý. 2. Hệ thống hiển thị mỗi yêu cầu với thông tin sinh viên, môn học/chủ đề, slot mong muốn, ghi chú của sinh viên. 3. Tutor chọn một yêu cầu để xem chi tiết mở rộng (lịch sử sinh viên, số lần hủy gần đây, ghi chú điều phối). 4. Tutor xác minh slot còn trống và kiểm tra xung đột với buổi khác. 5. Tutor chọn hành động “Chấp nhận”, “Từ chối” hoặc “Đề xuất lại thời gian”. 6. Nếu chấp nhận, Tutor sẽ thực hiện luồng trong use-case "Mở buổi tư vấn" từ bước 3, hệ thống gắn yêu cầu vào buổi tư vấn, chuyển trạng thái “Confirmed” và khóa slot. 7. Hệ thống tạo thông báo tới sinh viên, cập nhật dashboard.
Alternative Flow	A1 - Đề xuất thời gian khác: a. Ở bước 5, Tutor chọn “Đề xuất lại thời gian”. b. Hệ thống hiển thị các slot rảnh khác của Tutor; Tutor chọn slot mới và nhập lý do. c. Yêu cầu chuyển sang trạng thái “Reschedule Pending”, sinh viên nhận thông báo để xác nhận.
Postcondition	Mỗi yêu cầu được gán trạng thái mới (Approved/Declined/Rescheduled/Waitlisted) và cập nhật vào lịch Tutor và sinh viên. Thông báo tự động gửi tới sinh viên, điều phối viên (nếu cấu hình).
Exception Flow	Không có



Use-case ID	UC-TUTOR-006
Use-case	Theo dõi sinh viên
Actor	Tutor
Description	Cung cấp cho Tutor khả năng xem và cập nhật tiến độ học tập của từng mentee, bao gồm lịch sử buổi, phản hồi, và các hành động hỗ trợ cần triển khai, nhằm theo dõi hiệu quả chương trình tutor.
Precondition	1. Tutor đã đăng nhập hợp lệ qua HCMUT_SSO. 2. Tutor đã được gán nhóm mentee (từ ghép cặp hoặc điều phối viên). 3. Dữ liệu buổi tutor (đã diễn ra, sắp diễn ra) và phản hồi sinh viên được hệ thống lưu trữ.
Trigger	Tutor chọn chức năng “Theo dõi sinh viên” từ dashboard hoặc từ chi tiết buổi sau khi hoàn tất.
Normal Flow	1. Tutor mở giao diện “Theo dõi sinh viên”. 2. Hệ thống hiển thị danh sách mentee thuộc phạm vi Tutor phụ trách với các chỉ số tổng quan (số buổi đã tham gia, trạng thái phản hồi, cảnh báo). 3. Tutor chọn một mentee để xem chi tiết hồ sơ học tập. 4. Hệ thống hiển thị lịch sử buổi (đã diễn ra, sắp tới), phản hồi định lượng/định tính, ghi chú của Tutor trước đó, tài liệu liên quan. 5. Tutor phân tích thông tin, đánh dấu mức độ tiến bộ, thêm ghi chú hành động (ví dụ: “Cần ôn lại chương 3”, “Chuẩn bị bài tập thực hành cho buổi sau”). 6. Tutor có thể gắn thẻ cảnh báo hoặc gửi yêu cầu hỗ trợ bổ sung tới điều phối viên nếu phát hiện rủi ro (ví dụ: vắng 2 buổi liên tiếp). 7. Tutor lưu cập nhật, cập nhật dashboard và đồng bộ với module báo cáo.
Alternative Flow	Không có
Postcondition	1. Tutor có thể xem tổng quan tình trạng học tập của từng mentee (số buổi tham gia, trạng thái, điểm phản hồi). 2. Các ghi chú theo dõi, hành động follow-up được lưu lại và có thể chia sẻ với điều phối viên/khoa khi cần. 3. Hệ thống cập nhật báo cáo tiến độ phục vụ các bộ phận liên quan (khoa, phòng CTSV).
Exception Flow	E1. Dữ liệu phản hồi chưa có: Bước 4 phát hiện mentee chưa gửi phản hồi; hệ thống hiển thị cảnh báo và cung cấp nút “Gửi yêu cầu phản hồi lại”. E2. Ghi chú lưu thất bại: Sau bước 7, lưu ghi chú bị lỗi; hệ thống hiển thị cảnh báo, giữ nội dung thảo để Tutor thử lại sau, log sự cố cho đội vận hành.



Use-case ID	UC-TUTOR-007
Use-case	Dánh giá sinh viên
Actor	Tutor
Description	Cho phép Tutor ghi nhận mức độ tiến bộ, chất lượng tham gia buổi học và các khuyến nghị tiếp theo cho từng mentee sau buổi tutor, nhằm cung cấp dữ liệu cho báo cáo học tập và phòng CTSV.
Precondition	<ol style="list-style-type: none">Tutor đã đăng nhập hợp lệ qua HCMUT_SSO và có quyền truy cập mentee.Có ít nhất một buổi tutor đã diễn ra với mentee (trạng thái “Completed” hoặc “Awaiting Evaluation”).Dữ liệu buổi (ngày, chủ đề, mục tiêu) được lưu sẵn để liên kết với phiếu đánh giá.
Trigger	Tutor chọn chức năng “Dánh giá sinh viên” từ dashboard sau buổi học hoặc từ danh sách buổi cần đánh giá.
Normal Flow	<ol style="list-style-type: none">Tutor mở màn hình “Dánh giá sinh viên”.Hệ thống hiển thị danh sách mentee/buổi đang chờ đánh giá cùng các thông tin tóm tắt (ngày, chủ đề, trạng thái phản hồi).Tutor chọn mentee/buổi cần đánh giá để vào chi tiết.Hệ thống tải form đánh giá gồm mục định lượng (ví dụ: mức độ đạt mục tiêu 1-5) và phần nhận xét, gợi ý hành động tiếp theo.Tutor nhập điểm số, đánh dấu mức độ tiến bộ, ghi chú nhận xét, chọn gợi ý tài liệu bổ sung nếu cần.Tutor có thể đính kèm tài liệu minh chứng hoặc liên kết bài tập giao sau.Tutor xác nhận lưu; hệ thống kiểm tra dữ liệu bắt buộc (điểm, nhận xét tối thiểu) và thực hiện lưu vào kho đánh giá.
Alternative Flow	Không có
Postcondition	<ol style="list-style-type: none">Phiếu đánh giá được lưu với điểm định lượng (nếu dùng thang điểm) và nhận xét định tính, gắn với buổi và mentee tương ứng.Báo cáo tiến độ của mentee được cập nhật; điều phối viên/khoa có thể tra cứu.Nếu cấu hình, thông báo phản hồi được gửi cho mentee và ghi nhận vào hồ sơ điểm rèn luyện.
Exception Flow	<p>E1. Thiếu thông tin bắt buộc: Bước 7 phát hiện điểm hoặc nhận xét bị bỏ trống → hệ thống hiển thị thông báo lỗi, giữ dữ liệu đã nhập và yêu cầu bổ sung.</p> <p>E2. Lưu dữ liệu thất bại (lỗi hệ thống): Sau bước 7, việc ghi DB lỗi → hệ thống hiển thị thông báo “Không thể lưu, dữ liệu được tạm giữ”, ghi log và cho phép thử lại.</p>

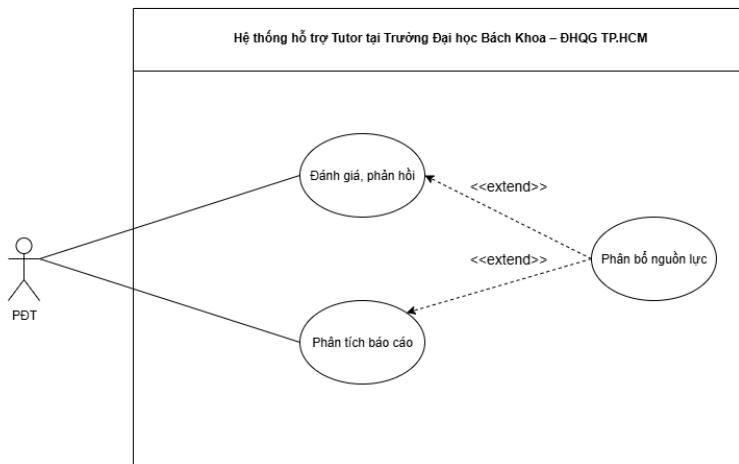


5.4 Khoa / Bộ môn



Use-case ID	UC-DEPT-001
Use-case	Theo dõi và đánh giá sinh viên
Actor	Khoa / Bộ môn
Description	Cho phép Khoa / Bộ môn truy cập dữ liệu đánh giá từ hệ thống để nắm tình hình học tập của sinh viên, từ đó hỗ trợ phân bổ nguồn lực giảng dạy và cải tiến chương trình đào tạo.
Precondition	Dữ liệu đánh giá từ các buổi tutor đã được ghi nhận đầy đủ.
Trigger	Cán bộ Khoa / Bộ môn truy cập mục “Báo cáo đánh giá sinh viên” từ dashboard.
Normal Flow	<ol style="list-style-type: none"> Người dùng mở giao diện báo cáo. Hệ thống hiển thị bộ lọc theo lớp, ngành, học kỳ. Người dùng chọn lớp/ngành cần xem. Hệ thống hiển thị bảng tổng hợp: số buổi tham gia, điểm phản hồi, đánh giá từ tutor. Người dùng xuất báo cáo để phục vụ phân tích nội bộ.
Alternative Flow	A1 - So sánh giữa các lớp: Người dùng chọn nhiều lớp để so sánh mức độ tham gia và kết quả học tập.
Postcondition	<ol style="list-style-type: none"> Báo cáo được trích xuất thành công. Khoa/Bộ môn có cơ sở để điều chỉnh phân bổ giảng viên hoặc nội dung môn học.
Exception Flow	E1 - Không có dữ liệu: Hệ thống hiển thị thông báo “Chưa có dữ liệu cho lớp/ngành này”. E2 - Lỗi xuất báo cáo: Hệ thống không tạo được file, hiển thị lỗi và cho phép thử lại.

5.5 Phòng Đào tạo





Use-case ID	UCPDT1
Use-case	Phân bổ nguồn lực
Actor	Phòng đào tạo
Description	PDT phân bổ tutor phù hợp cho sinh viên dựa trên hồ sơ, nhu cầu hoặc gợi ý từ hệ thống
Precondition	Sinh viên đã tham gia đăng ký Tutor. Thông tin tutor và sinh viên đã được đồng bộ từ HCMUT_DATACORE.
Postcondition	Sinh viên được gán tutor thành công, dữ liệu cập nhật vào hệ thống.
Trigger	PDT cần phân công tutor cho sinh viên mới đăng ký.
Normal Flow	1. PDT đăng nhập hệ thống qua HCMUT_SSO. 2. Hệ thống hiển thị danh sách sinh viên chưa có tutor hoặc muốn đổi tutor. 3. PDT chọn một sinh viên. 4. Hệ thống gợi ý danh sách tutor phù hợp (dựa trên chuyên môn, lịch rảnh, số lượng mentee hiện có). 5. Hệ thống xác nhận, gửi thông báo đến tutor và sinh viên.
Exception Flow	Không có tutor phù hợp → Hệ thống báo lỗi và yêu cầu PDT xử lý sau.
Alternative Flow	PDT có thể bỏ qua gợi ý và chọn tutor thủ công.

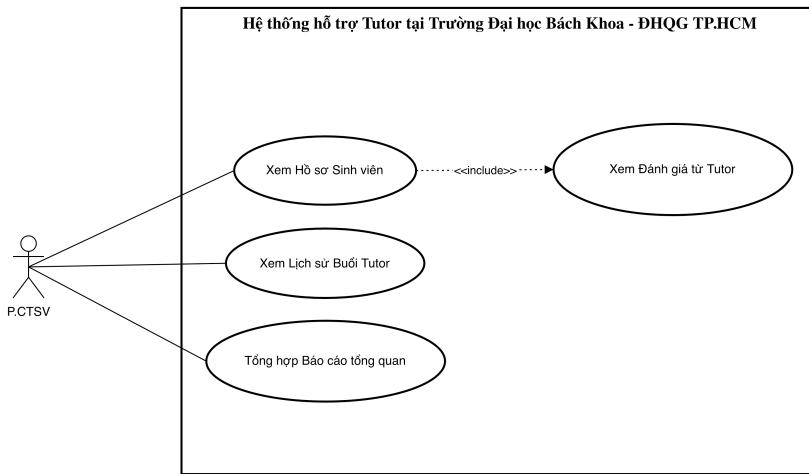
Use-case ID	UCPDT2
Use-case	Phân tích báo cáo
Actor	Phòng đào tạo
Description	PDT xem báo cáo tổng quan về tình hình hoạt động của chương trình Tutor.
Precondition	Hệ thống đã lưu trữ dữ liệu buổi gấp, phản hồi, tiến độ học tập.
Postcondition	Báo cáo được tạo thành công.
Trigger	PDT cần báo cáo định kỳ hoặc đột xuất.
Normal Flow	1. PDT đăng nhập hệ thống. 2. Chọn chức năng “Phân tích báo cáo”. 3. Hệ thống hiển thị các tùy chọn: theo khoa, theo môn học, theo tutor, theo kỳ học. 4. PDT chọn phạm vi báo cáo. 5. Hệ thống tạo báo cáo và hiển thị biểu đồ/tài liệu. 6. PDT có thể tải xuống hoặc in báo cáo.
Exception Flow	Dữ liệu chưa đủ hoặc lỗi đồng bộ → Hệ thống thông báo và đề nghị thử lại.



Alternative Flow	PDT lọc báo cáo theo nhiều tiêu chí kết hợp (ví dụ: khoa + học kỳ).
-------------------------	---

Use-case ID	UCPDT3
Use-case	Dánh giá phản hồi
Actor	Phòng đào tạo
Description	PDT khai thác phản hồi từ sinh viên và tutor để đánh giá chất lượng chương trình.
Precondition	Sinh viên và tutor đã gửi phản hồi sau buổi học.
Postcondition	PDT có dữ liệu đánh giá để cải tiến chất lượng chương trình.
Trigger	PDT cần phân tích chất lượng chương trình Tutor.
Normal Flow	1. PDT đăng nhập hệ thống. 2. Chọn chức năng “Dánh giá phản hồi”. 3. Hệ thống hiển thị danh sách phản hồi theo buổi học, theo tutor, hoặc theo sinh viên. 4. PDT xem chi tiết phản hồi. 5. PDT ghi nhận kết quả để cải tiến chương trình.
Exception Flow	Không có phản hồi nào được ghi nhận → Hệ thống hiển thị thông báo “Chưa có dữ liệu”.
Alternative Flow	PDT lọc phản hồi theo thời gian hoặc theo nhóm sinh viên đặc biệt.

5.6 Phòng Công tác sinh viên



Use-case ID	UC-PCT-01
--------------------	-----------

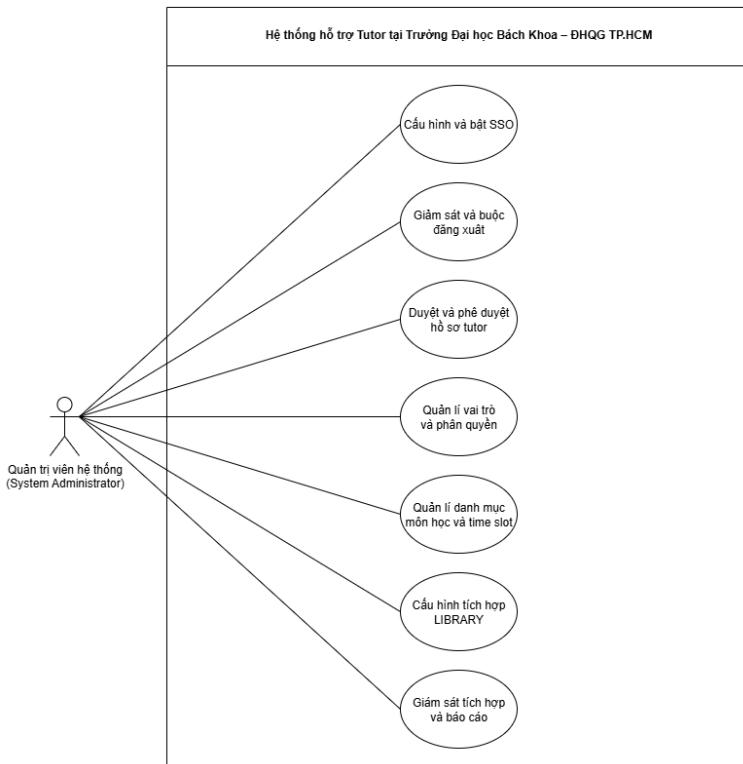
Use-case	Xem Hồ sơ Sinh viên
Actor	Cán bộ quản lý của P.CTSV
Description	Cho cán bộ xem thông tin chi tiết về hồ sơ cá nhân của sinh viên
Trigger	Người dùng chọn chức năng 'Xem Hồ sơ Sinh viên'.
Precondition	- Hệ thống BKArch hoạt động bình thường. - Người dùng đã đăng nhập thành công vào hệ thống Tutor.
Postcondition	- Thành công: Hiển thị thông tin hồ sơ sinh viên chi tiết.
Normal Flow	1. Người dùng chọn chức năng 'Xem Hồ sơ Sinh viên'. 2. Hệ thống truy xuất dữ liệu hồ sơ sinh viên. 3. Hệ thống hiển thị thông tin toàn bộ hồ sơ sinh viên. 4. Người dùng chọn hồ sơ một sinh viên cụ thể để xem thông tin. 5. Hệ thống hiển thị hồ sơ chi tiết của sinh viên.
Exception Flow	E1. Lỗi kết nối CSDL: Hệ thống ghi lại log, hiển thị lỗi.
Alternative Flow	A1. Lọc danh sách: a. Tại bước 3, người dùng lọc ra hồ sơ sinh viên dựa trên MSSV, Học Kỳ, b. Hệ thống hiển thị các hồ sơ đã được lọc.

Use-case ID	UC-PCT-02
Use-case	Xem Ghi nhận Tiền bộ từ Tutor
Actor	Cán bộ quản lý của P.CTSV
Description	Giảng viên hoặc Cán bộ quản lý đã ghi nhận tiền bộ cho sinh viên.
Trigger	Người dùng truy cập chi tiết một 'Hồ sơ Sinh viên'.
Precondition	- Hệ thống BKArch đang hoạt động. - Người dùng đã đăng nhập thành công. - Người dùng đang xem hồ sơ của một sinh viên cụ thể.
Postcondition	- Thành công: Hiển thị danh sách các ghi nhận tiền bộ của sinh viên.
Normal Flow	1. Người dùng truy cập 'Chi tiết' khi đang xem 1 'Hồ sơ Sinh viên' (UC-PCT-01). 2. Hệ thống truy xuất và hiển thị danh sách ghi nhận tiền bộ liên quan đến hồ sơ đó.
Exception Flow	E1. Lỗi kết nối CSDL: Hệ thống ghi lại log và hiển thị lỗi.
Alternative Flow	A1. Lọc, sắp xếp danh sách Hồ sơ Sinh viên: a. Người dùng lọc hoặc sắp xếp dựa trên Môn học, Điểm số, ... b. Hệ thống hiển thị danh sách đã được lọc.

Use-case ID	UC-PCT-03
Use-case	Xem Lịch sử Buổi Tutor
Actor	Cán bộ quản lý của P.CTSV
Description	Cho phép người dùng xem lại lịch sử chi tiết các buổi Tutor mà sinh viên đã tham gia hoặc được ghi nhận.
Trigger	Người dùng chọn chức năng 'Lịch sử Buổi Tutor'.
Precondition	- Hệ thống BKArch đang hoạt động. - Người dùng đã đăng nhập thành công.
Postcondition	- Thành công: Hiển thị danh sách và chi tiết các buổi Tutor đã diễn ra.
Normal Flow	1. Người dùng chọn chức năng 'Xem Lịch sử Buổi Tutor'. 2. Hệ thống truy xuất dữ liệu các buổi Tutor của sinh viên. 3. Hệ thống hiển thị danh sách và chi tiết các buổi Tutor.
Exception Flow	E1. Lỗi kết nối CSDL: Hệ thống ghi log và hiển thị lỗi.
Alternative Flow	A1. Lọc danh sách Buổi Tư vấn: a. Người dùng lọc ra các buổi Tutor dựa trên Học Kỳ, Môn Học, ... b. Hệ thống hiển thị danh sách buổi Tutor đã được lọc.

Use-case ID	UC-PCT-04
Use-case	Tổng hợp Báo cáo Cộng Điểm Rèn luyện
Actor	Sinh viên
Description	Cho phép người dùng tổng hợp và xem báo cáo về việc cộng điểm rèn luyện của sinh viên.
Trigger	Người dùng chọn chức năng 'Tổng hợp Báo cáo Tổng quan'.
Precondition	- Hệ thống BKArch đang hoạt động. - Người dùng đã đăng nhập thành công. - Dữ liệu về danh sách buổi Tư vấn đã được cập nhật đầy đủ.
Postcondition	- Thành công: Hiển thị báo cáo tổng hợp về cộng điểm rèn luyện.
Normal Flow	1. Người dùng chọn chức năng 'Tổng hợp Báo cáo Cộng Điểm Rèn luyện'. 2. Hệ thống thực hiện tổng hợp dữ liệu. 3. Hệ thống hiển thị báo cáo.
Exception Flow	E1. Lỗi trong quá trình tổng hợp dữ liệu: Hệ thống ghi log và hiển thị lỗi. E2. Lỗi kết nối CSDL: Hệ thống ghi log và hiển thị lỗi.
Alternative Flow	A1. Không có dữ liệu điểm rèn luyện để tổng hợp: Hệ thống hiển thị danh sách trống.

5.7 Quản trị viên hệ thống (System Administrator)



Use-case ID	UCSA1
Use-case	Cấu hình & bật HCMUT_SSO
Actor	SysAdmin
Description	Cấu hình và bật tính năng đăng nhập hợp nhất qua HCMUT_SSO.
Precondition	Có client_id, client_secret, redirect URI đã whitelist.
Postcondition	Người dùng có thể đăng nhập bằng tài khoản HCMUT_SSO.
Trigger	SysAdmin cần bật đăng nhập hợp nhất để triển khai hệ thống.
Normal Flow	<ol style="list-style-type: none"> SysAdmin đăng nhập vào hệ thống quản trị. Chọn menu System Settings → Authentication → SSO Integration. Nhập client_id, client_secret, redirect URI, discovery URL. Nhấn Test Connection. Hệ thống gửi request đến HCMUT_SSO và nhận về JWKS. Hệ thống hiển thị kết quả “Connection Successful”. SysAdmin nhấn Save & Enable để bật SSO.



Exception Flow	E1: Sai client_id hoặc client_secret → Hệ thống báo <i>Invalid credentials</i> . E2: Timeout kết nối → Hiển thị banner <i>SSO Service not reachable</i> . E3: Lỗi lưu cấu hình → Rollback và báo <i>Cannot save configuration</i> .
Alternative Flow	Không có.

Use-case ID	UCSA2
Use-case	Giám sát & buộc đăng xuất
Actor	SysAdmin
Description	Theo dõi tình trạng phiên đăng nhập và buộc logout khi cần.
Precondition	SSO đã được bật.
Postcondition	Phiên người dùng bị ngắt, trạng thái được cập nhật trên dashboard.
Trigger	SysAdmin cần chấm dứt phiên rủi ro.
Normal Flow	1. SysAdmin mở Monitoring → Active Sessions. 2. Dashboard hiển thị danh sách user đang đăng nhập. 3. SysAdmin lọc user theo tên/email/khoa. 4. Chọn một hoặc nhiều user. 5. Nhấn Force Logout. 6. Hệ thống hiển thị hộp thoại xác nhận → SysAdmin đồng ý. 7. Hệ thống gửi lệnh huỷ session tới SSO. 8. Log lưu lại hành động.
Exception Flow	E1: Mất kết nối với SSO → Banner <i>SSO temporarily unavailable</i> . E2: SysAdmin không đủ quyền → Báo lỗi <i>Permission Denied</i> .
Alternative Flow	Không có.

Use-case ID	UCSA3
Use-case	Duyệt & phê duyệt hồ sơ tutor
Actor	SysAdmin
Description	Quản lý quá trình duyệt hồ sơ tutor trước khi hiển thị công khai.
Precondition	Tutor đã submit hồ sơ (profile, chứng chỉ, minh chứng).
Postcondition	Tutor được approve thì xuất hiện trong kết quả tìm kiếm, reject thì lưu lý do.
Trigger	Có tutor gửi hồ sơ chờ duyệt.



Normal Flow	1. SysAdmin vào User Management → Tutors → Pending Applications. 2. Hệ thống hiển thị danh sách tutor chờ duyệt. 3. SysAdmin chọn một hồ sơ → xem chi tiết. 4. Nếu hợp lệ → nhấn Approve. 5. Nếu chưa đủ → nhấn Reject, nhập lý do. 6. Hệ thống gửi email thông báo cho tutor. 7. Nếu approved → hệ thống gán role tutor và hiển thị.
Exception Flow	E1: Hồ sơ thiếu minh chứng → Cảnh báo <i>Missing documents</i> . E2: Lỗi cập nhật DB → Rollback, báo <i>Failed to update status</i> .
Alternative Flow	SysAdmin có thể gửi Request More Info thay vì Reject.

Use-case ID	UCSA4
Use-case	Quản lý role & phân quyền
Actor	SysAdmin
Description	Thiết lập quy tắc ánh xạ claim → role hệ thống.
Precondition	Hệ thống nhận claim từ SSO/DATACORE.
Postcondition	Role được cập nhật tự động mỗi lần login hoặc refresh.
Trigger	Admin cần thiết lập quy tắc RBAC.
Normal Flow	1. SysAdmin mở Access Control → Role Mapping Rules. 2. Nhấn Add New Rule. 3. Chọn nguồn claim. 4. Nhập điều kiện. 5. Chọn role ánh xạ. 6. Nhấn Test Rule với user mẫu. 7. Nhấn Publish để áp dụng.
Exception Flow	E1: Claim không tồn tại → Cảnh báo <i>Claim missing</i> , gán mặc định student. E2: Xung đột rule → Hiển thị mâu thuẫn, yêu cầu sắp xếp ưu tiên.
Alternative Flow	Import/Export rule dưới dạng JSON/YAML.

Use-case ID	UCSA5
Use-case	Quản lý danh mục môn học & timeslot
Actor	SysAdmin
Description	Quản trị danh mục môn học, học kỳ và khung giờ học.
Precondition	Hệ thống có module catalog.
Postcondition	Danh mục môn học & timeslot luôn đồng bộ.



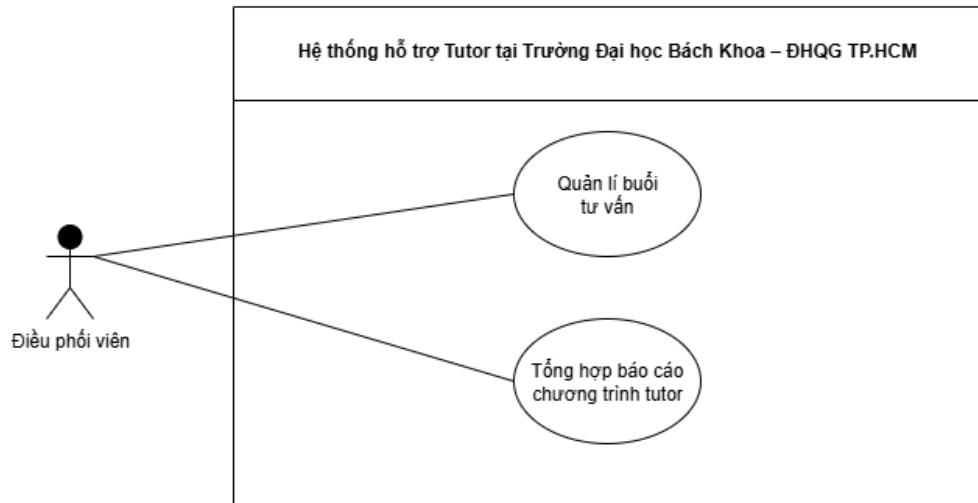
Trigger	Có nhu cầu cập nhật danh mục.
Normal Flow	1. SysAdmin mở Catalog → Subjects/Timeslots. 2. Nhấn Add New. 3. Nhập thông tin môn học/timeslot. 4. Nhấn Save. 5. Danh mục cập nhật ngay cho sinh viên/tutor. 6. Với dữ liệu đã tham chiếu → chỉ cho phép Archive.
Exception Flow	E1: Trùng mã môn học → Báo <i>Duplicate Subject Code</i> . E2: Xoá môn đang dùng → Chặn xoá, yêu cầu Archive.
Alternative Flow	Không có.

Use-case ID	UCSA6
Use-case	Giám sát tích hợp & báo cáo
Actor	SysAdmin
Description	Theo dõi sức khỏe hệ thống và xuất báo cáo tích hợp.
Precondition	Các tích hợp đã bật (SSO, DATACORE, LIBRARY).
Postcondition	Báo cáo đầy đủ giúp giám sát, audit minh bạch.
Trigger	Admin cần giám sát hệ thống.
Normal Flow	1. SysAdmin mở Reports → Integration Dashboard. 2. Hệ thống hiển thị KPI: login SSO, sync/day, mapping errors, library access. 3. SysAdmin lọc theo thời gian, nguồn. 4. Nhấn KPI bất thường để xem log chi tiết. 5. Xuất CSV/PDF. 6. Audit log việc export.
Exception Flow	E1: Job sync fail → Banner đỏ + link “View Logs”. E2: Thiếu quyền Export → Báo <i>Restricted Access</i> .
Alternative Flow	Đặt lịch gửi báo cáo định kỳ.

Use-case ID	UCSA7
Use-case	Cấu hình tích hợp HCMUT_LIBRARY
Actor	SysAdmin
Description	Kết nối hệ thống với HCMUT_LIBRARY để quản lý tài liệu tutor.
Precondition	Có OAuth/API key hợp lệ từ LIBRARY.
Postcondition	Người dùng truy cập tài liệu đúng quyền.
Trigger	Admin cần tích hợp LIBRARY.

Normal Flow	1. SysAdmin vào Integrations → Library. 2. Nhập OAuth/API key. 3. Chọn scope cho từng role. 4. Nhấn Save. 5. Chạy Test Access với user mẫu → hiển thị OK.
Exception Flow	E1: Token hết hạn → Tự refresh; nếu fail → cảnh báo. E2: Scope không hợp lệ → Báo <i>Invalid Scope</i> .
Alternative Flow	Không có.

5.8 Điều phối viên



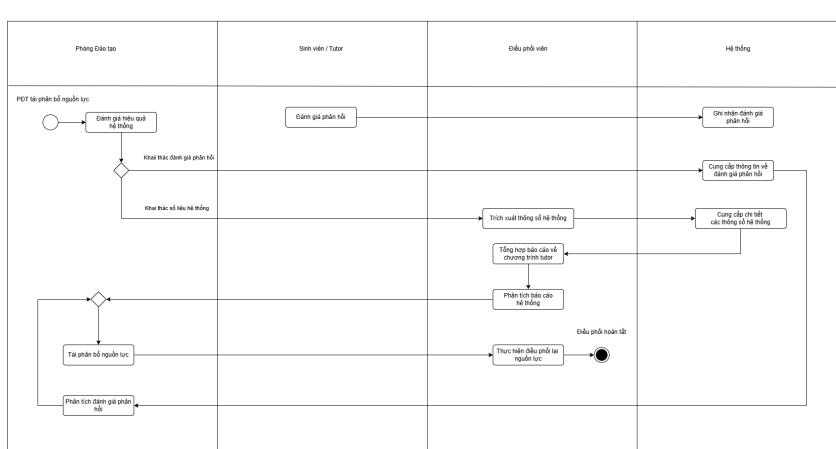
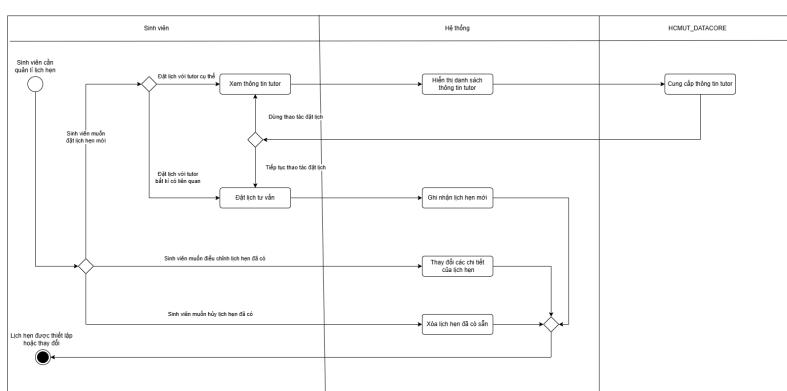
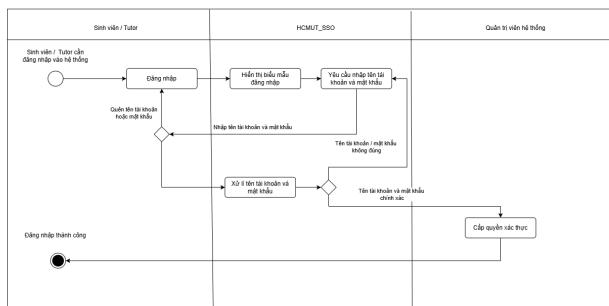


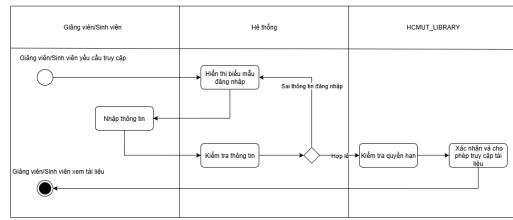
Use-case ID	UC-COORD-001
Use-case	Quản lý buổi tư vấn
Actor	Điều phối viên
Description	Cho phép điều phối viên theo dõi tiến độ chương trình tutor, phân bổ tutor cho nhóm sinh viên, xử lý các vấn đề phát sinh trong quá trình triển khai buổi tư vấn.
Precondition	Dữ liệu nhóm sinh viên, lịch tutor, trạng thái buổi đã được cập nhật.
Trigger	Điều phối viên truy cập dashboard quản lý chương trình tutor.
Normal Flow	<ol style="list-style-type: none">Điều phối viên mở giao diện quản lý chương trình.Hệ thống hiển thị danh sách nhóm sinh viên, trạng thái tutor, số buổi đã diễn ra.Điều phối viên phân bổ tutor cho nhóm chưa có người phụ trách.Điều phối viên theo dõi tiến độ từng nhóm, xem cảnh báo học tập.Nếu phát sinh sự cố (tutor nghỉ, sinh viên phản hồi tiêu cực), điều phối viên xử lý hoặc thay thế.
Alternative Flow	A1 - Ghép lại nhóm: Nếu nhóm có quá ít sinh viên, điều phối viên có thể gộp nhóm và phân bổ lại tutor. A2 - Tạm khóa buổi: Nếu buổi có vấn đề, điều phối viên có thể khóa trạng thái để chờ xử lý.
Postcondition	<ol style="list-style-type: none">Các nhóm sinh viên được phân bổ tutor đầy đủ.Các sự cố phát sinh được ghi nhận và xử lý.
Exception Flow	E1 - Thiếu tutor: Hệ thống cảnh báo “Không đủ tutor để phân bổ”, điều phối viên cần gửi yêu cầu tuyển bổ sung. E2 - Lỗi cập nhật trạng thái: Hệ thống không ghi nhận thay đổi, hiển thị lỗi và cho phép thử lại.

Use-case ID	UC-COORD-002
Use-case	Tổng hợp báo cáo chương trình tutor
Actor	Điều phối viên
Description	Cho phép điều phối viên tổng hợp báo cáo định kỳ về tiến độ chương trình tutor, mức độ tham gia, chất lượng phản hồi để gửi tới các đơn vị liên quan như PDT, Khoa, CTSV.
Precondition	Dữ liệu buổi tutor, phản hồi sinh viên, đánh giá từ tutor đã được ghi nhận đầy đủ.
Trigger	Điều phối viên truy cập mục “Tổng hợp báo cáo” từ dashboard.
Normal Flow	<ol style="list-style-type: none"> 1. Điều phối viên mở giao diện báo cáo. 2. Hệ thống hiển thị bộ lọc theo học kỳ, ngành, nhóm tutor. 3. Điều phối viên chọn phạm vi cần báo cáo. 4. Hệ thống tổng hợp số liệu: số buổi đã diễn ra, tỷ lệ tham gia, cảnh báo học tập, mức độ hài lòng. 5. Điều phối viên xuất báo cáo PDF hoặc Excel. 6. Báo cáo được gửi tới các đơn vị liên quan.
Alternative Flow	<p>A1 - Báo cáo theo lớp: Điều phối viên chọn lớp cụ thể để tạo báo cáo chi tiết.</p> <p>A2 - Báo cáo theo nhóm tutor: Điều phối viên lọc theo từng tutor để đánh giá hiệu quả.</p>
Postcondition	<ol style="list-style-type: none"> 1. Báo cáo được trích xuất thành công. 2. Các đơn vị liên quan nhận được dữ liệu để phục vụ quản lý và cải tiến chương trình.
Exception Flow	<p>E1 - Lỗi xuất báo cáo: Hệ thống không tạo được file, hiển thị lỗi và cho phép thử lại.</p> <p>E2 - Thiếu dữ liệu: Hệ thống cảnh báo “Dữ liệu chưa đầy đủ để tạo báo cáo”.</p>

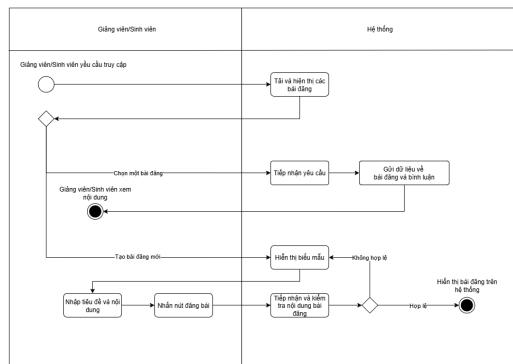
6 Lược đồ hoạt động và trình tự (Activity and Sequence Diagram)

6.1 Lược đồ hoạt động

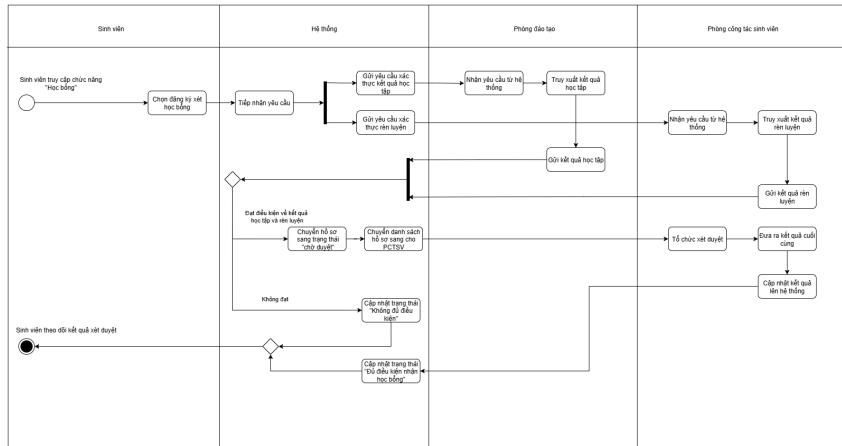




Activity diagram: Quản lý học liệu

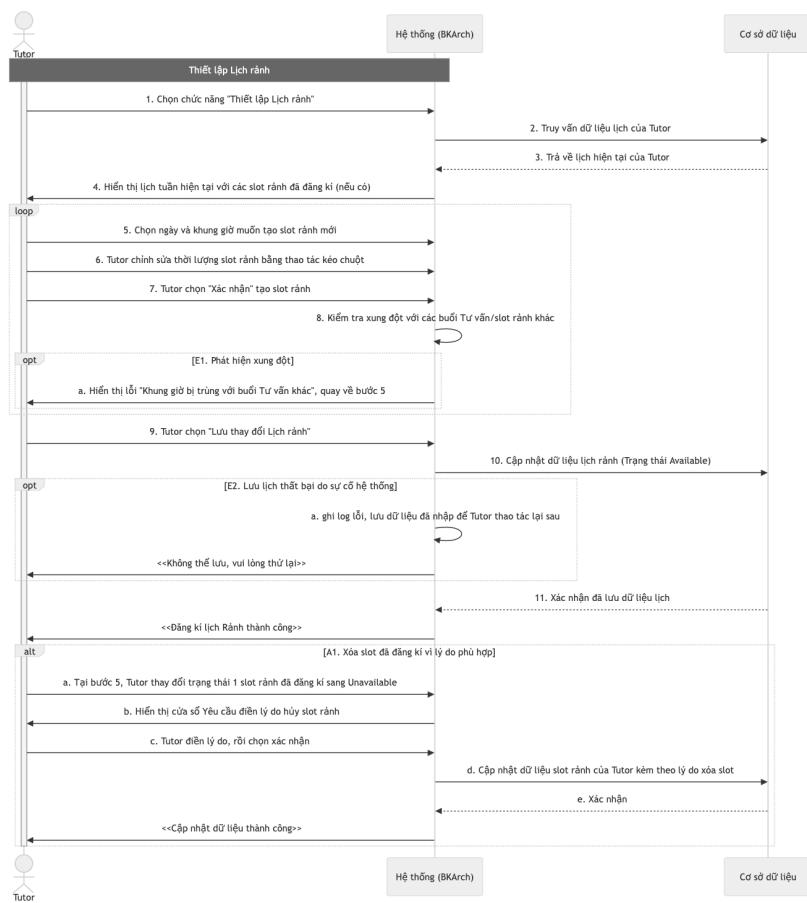


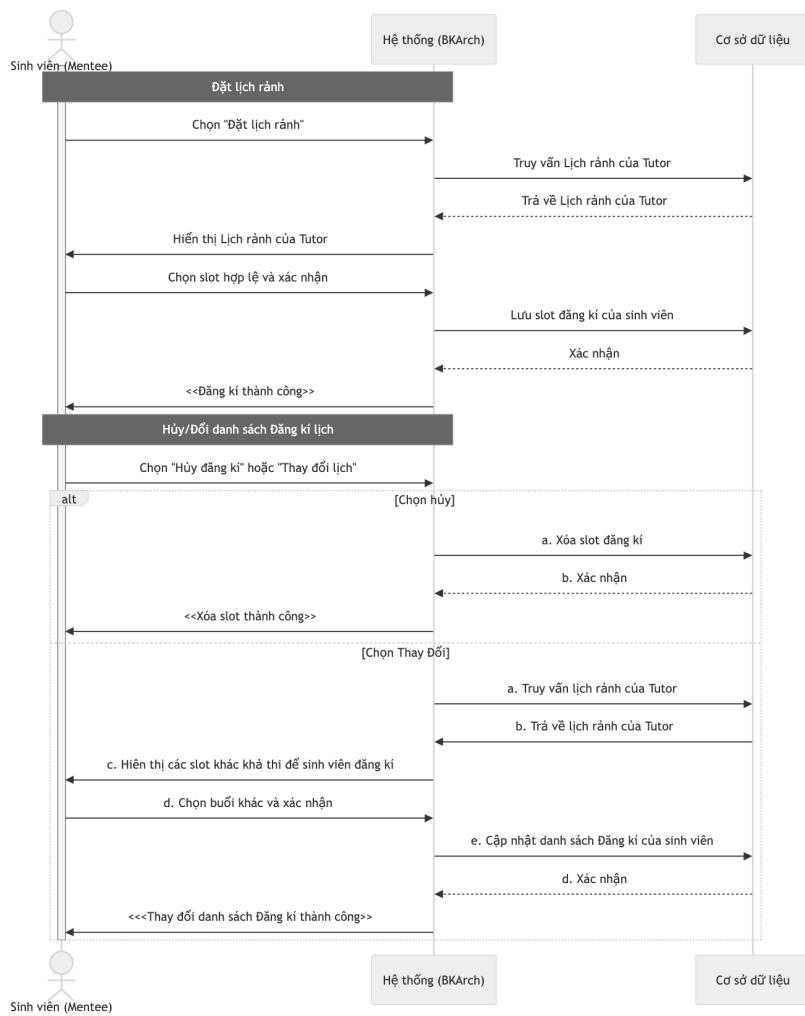
Activity diagram: Diễn đàn môn học

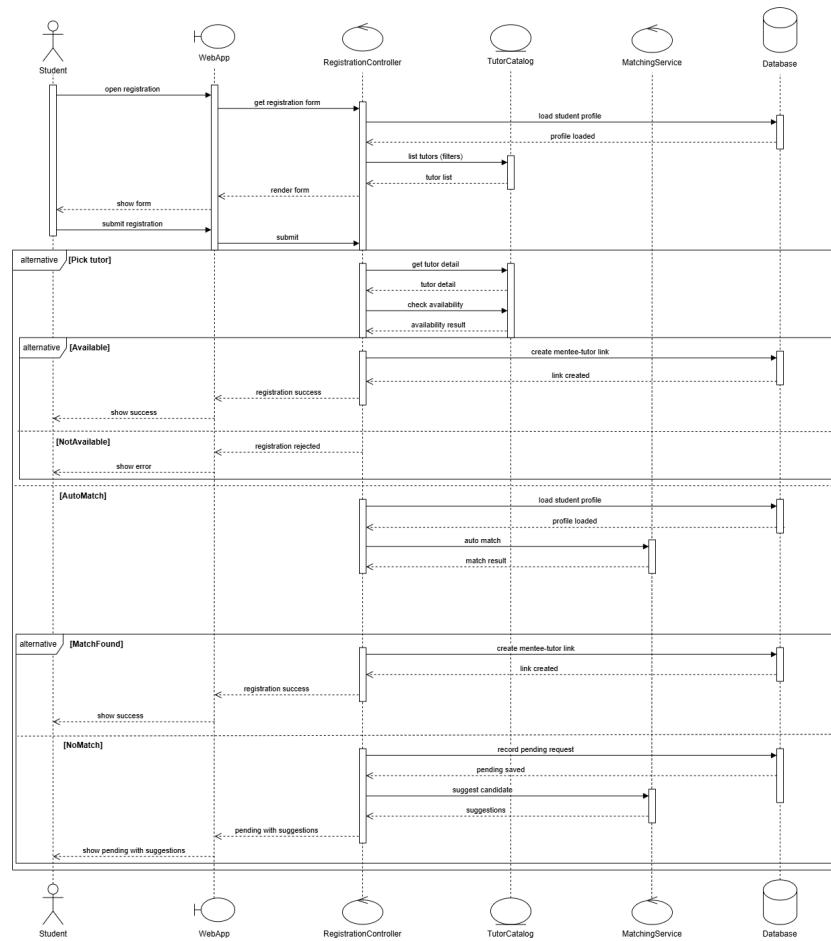


Activity diagram: Xét duyệt học bổng

6.2 Lược đồ trình tự

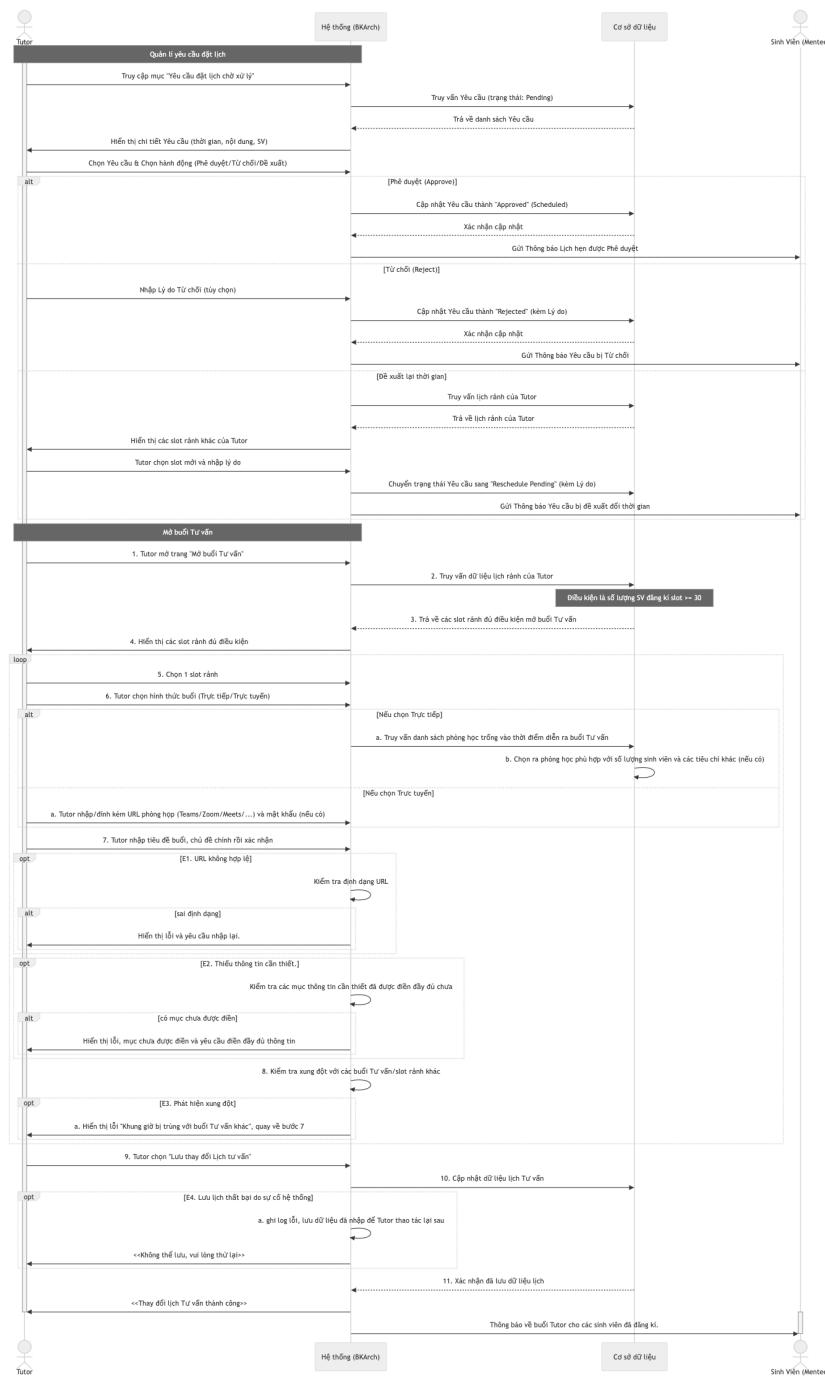


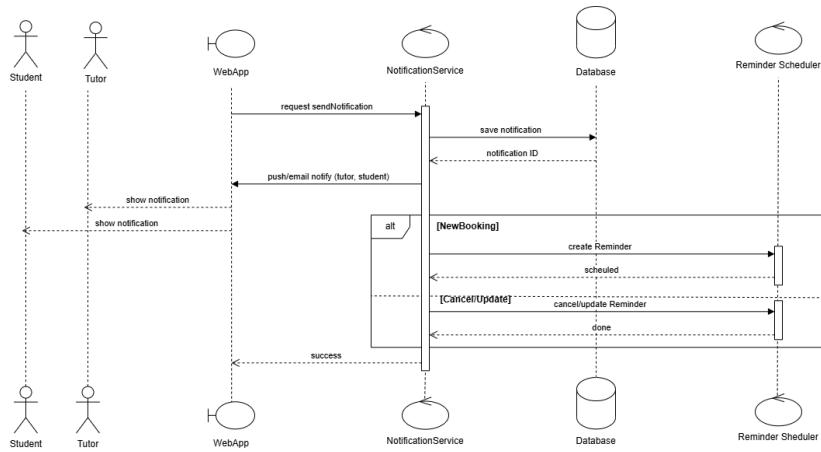


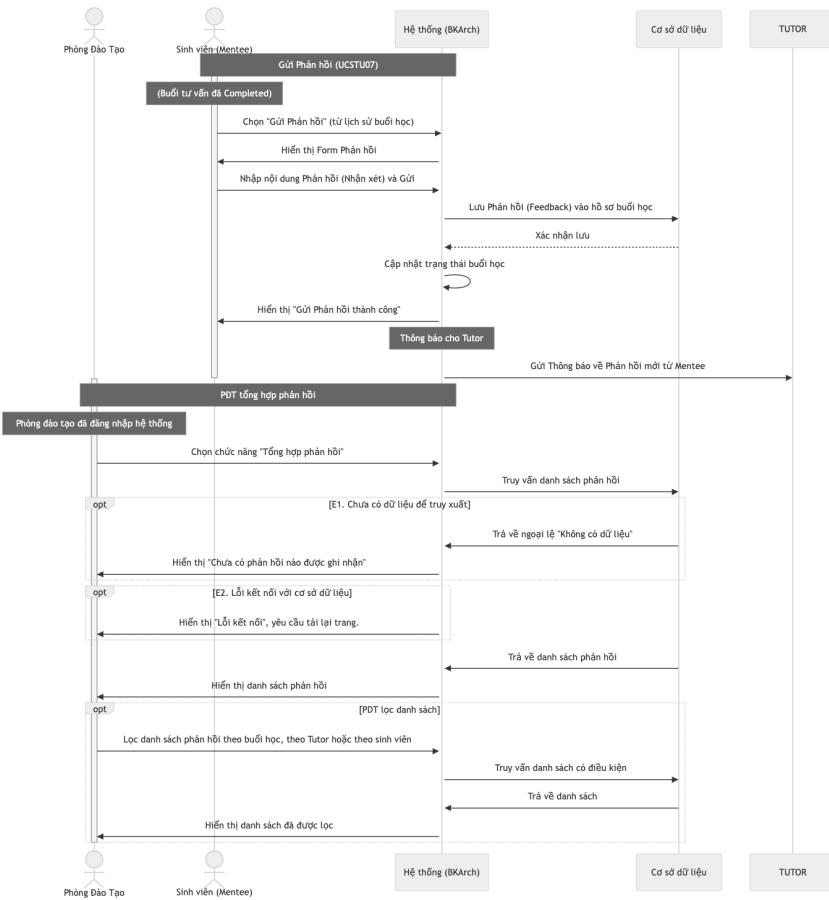


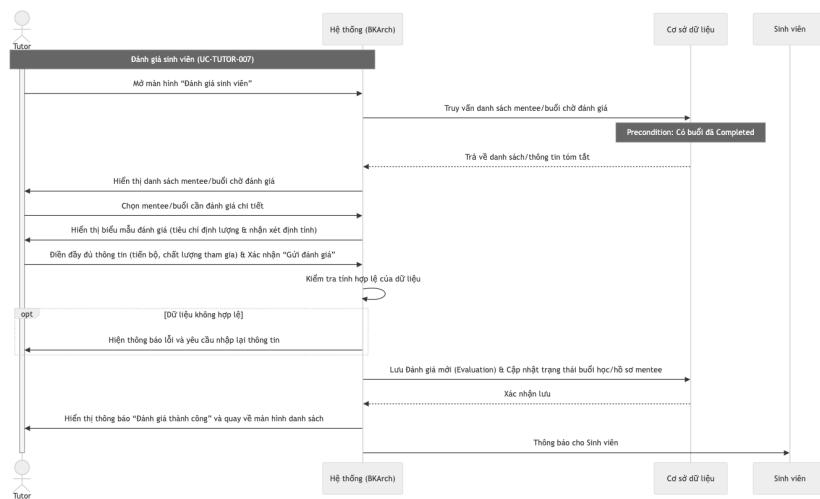


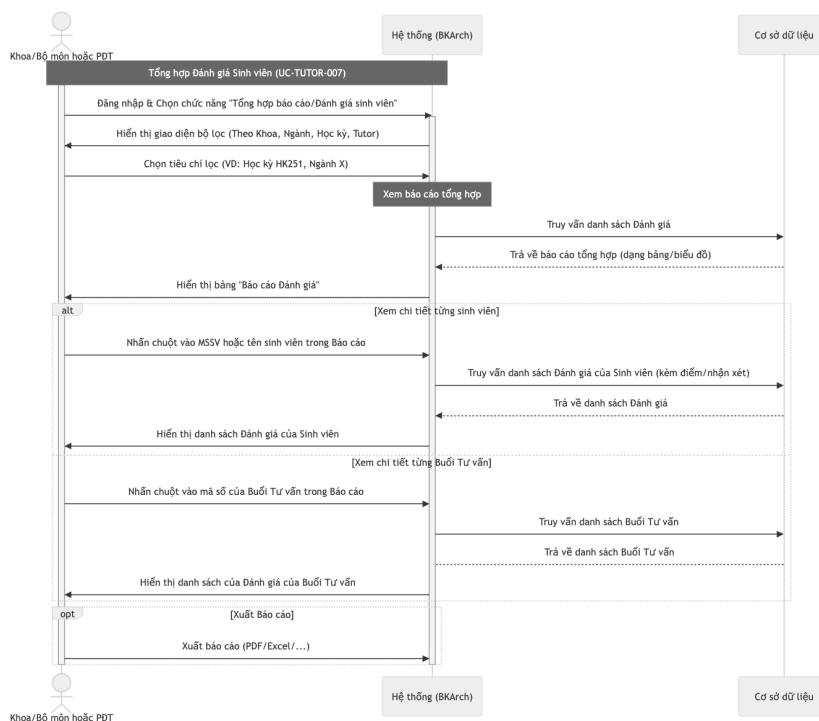
University of Technology, Ho Chi Minh City
Faculty of Computer Science and Engineering

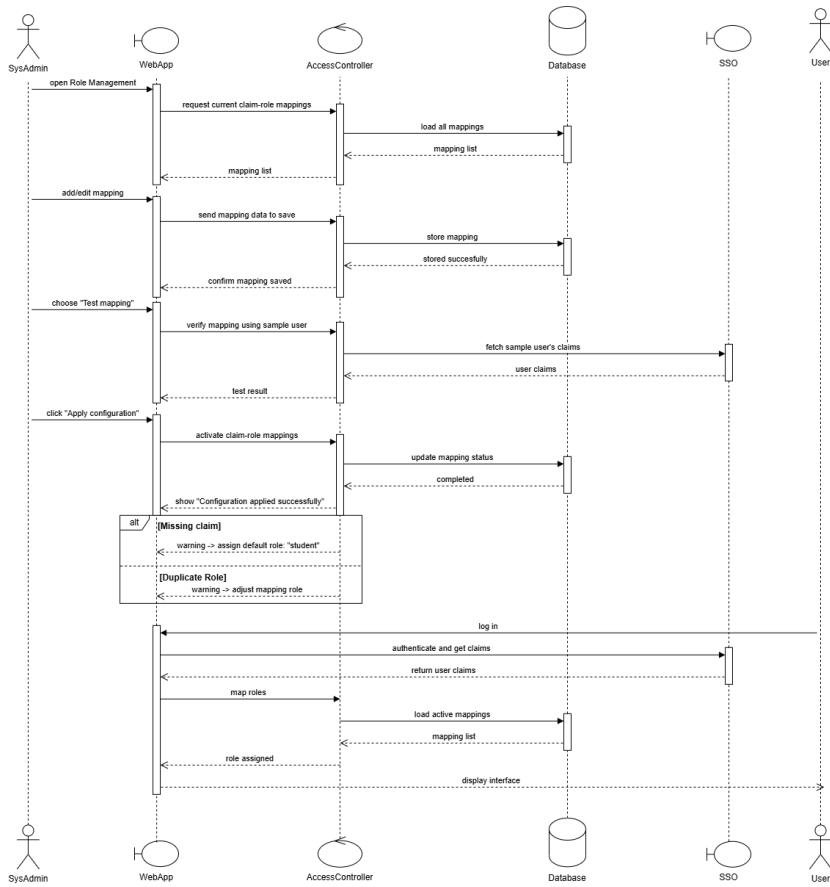














7 Mockup-UI

7.1 Sinh viên

The image displays four screenshots of the BK Tutor student application interface:

- Tutor Registration Screen:** Shows a search bar and three tutor profiles: Tutor1 (Mô tả tutor1), Tutor2 (Mô tả tutor2), and Tutor3 (Mô tả tutor3). Each profile has a "Read more >" link.
- Schedule View:** Shows a weekly calendar grid from Monday to Sunday. A tooltip for a specific slot indicates it is reserved for "Công nghệ phần mềm" (Software Engineering) at 10AM on Tuesday.
- Class History View:** Shows a card for "Công Nghệ Phần Mềm" (Software Engineering) with details: Thầy Mai Đức Trung, Thứ Hai: 10AM - 1PM, Địa điểm: meet.google.com/abc-xyz-mnp. Buttons for "Hủy lịch" (Cancel) and "Đổi lịch" (Change) are present.
- Class History View (Details):** Shows a detailed view of the "Công Nghệ Phần Mềm" class entry, including the teacher's name, day, time, location, and a note "Thay đổi lịch..." (Change schedule...).



University of Technology, Ho Chi Minh City

Faculty of Computer Science and Engineering

Ngôn ngữ: English

Đăng ký tài liệu

Chia sẻ tài liệu

Chia sẻ tài liệu mới ban cần và chia sẻ tài liệu ban cũ cho người khác nhau

Tài liệu

Loại tài liệu

Tên người đăng

Date

Công nghệ phần mềm

Đề tài

Nguyễn Thị Hải

Jan 1, 2020

View

Kỹ thuật lập trình

Tài liệu/lý thuyết

Phan Huy Quang Minh

Jan 11, 2020

View

Cử triết dữ liệu và giải thuật

Tài liệu/lý thuyết

Nguyễn Thị Hải

Jan 11, 2020

View

Mô hình hóa toán học

Tài liệu/lý thuyết

Ph Huân Quyết Mạng

Jan 11, 2020

View

Nguyên lý ngôn ngữ lập trình

Tài liệu/lý thuyết

Mark Zuckerberg

Jan 11, 2020

View

Công nghệ phần mềm

Tài liệu/lý thuyết

Evan Musk

Jan 11, 2020

View

< Prev

1 2 3 4

Next >

Đăng kí tham gia các seminar sắp tới

Tham gia seminar là cơ hội để mọi người kết nối, cập nhật các tin tức mới và giao lưu trao đổi chuyên đề công nghệ của bạn sinh viên công nghệ. Qua những buổi chia sẻ, bạn có thể luyện kỹ năng tư duy, thảo luận và kết nối – từ đó phát triển kỹ năng giao tiếp, chuẩn bị tốt cho tương lai.

30/02/2026

Hội thảo công nghệ

Hội thảo công nghệ là dịp để bạn khám phá những xu hướng, giải pháp và đổi mới nhất trong lĩnh vực công nghệ. Tại đây bạn sẽ được nghe các chuyên gia chia sẻ kiến thức chuyên sâu, tham gia thảo luận, trải nghiệm các công nghệ mới và kết nối với những người cùng đam mê. Tham gia hội thảo sẽ giúp bạn mở rộng kiến thức, nâng cao kỹ năng và chuẩn bị tốt hơn cho các cơ hội học tập và sự nghiệp trong tương lai.

Đăng kí Xem thêm >

31/4/2026

Seminar an toàn thông tin

Ngôn ngữ: English

Tham gia cộng đồng trực tuyến

Kết nối với mọi người cùng ngành, mở rộng mạng lưới kết nối với mọi người

Cộng đồng sinh viên khoa Máy tính

Cộng đồng dành cho các bạn sinh viên khoa máy tính

Cộng đồng sinh viên khoa Công nghệ

Cộng đồng dành cho các bạn sinh viên khoa ứng dụng

Cộng đồng sinh viên học cải thiện

Cộng đồng dành cho các bạn muốn học cải thiện công nghệ

Cộng đồng

← →



7.2 Tutor



University of Technology, Ho Chi Minh City

Faculty of Computer Science and Engineering

The screenshot shows a library catalog interface. On the left, there is a search bar and a list of books. Each book entry includes a thumbnail, title, author, publication year, and a 'Tìm kiếm' button. On the right, there is a sidebar with sections for 'Môn học của tôi' (My subjects) and 'Công Nghệ Phần Mềm' (Software Engineering), each with a 'Chi tiết' button.

This screenshot shows a list of scheduled events for 'Nguyễn Lý Ngôn Ngữ Lập Trình'. It includes three entries: 'OOP - Buổi 1' (Thứ Sáu 29 Tháng 10 2025), 'Ôn tập giữa kì' (Thứ Hai 03 Tháng 11 2025), and 'Jasmin code' (Thứ Năm 07 Tháng 11 2025). Each entry has a 'Sinh viên' button and a 'Chỉnh sửa' dropdown menu.

This screenshot shows the same course schedule as above, but with a 'Thay đổi thời gian' (Change time) dialog box open over the 'Jasmin code' entry. The dialog allows changing the start and end times from 9:00 to 10:00 to 13:00 to 14:00, with a 'Xác nhận' (Confirm) button.

This screenshot shows the course schedule after the time change has been confirmed. The 'Jasmin code' entry now shows the new time slot of 13:00 - 14:00. A 'Kết thúc' (End) button is visible at the bottom of the dialog.



University of Technology, Ho Chi Minh City
Faculty of Computer Science and Engineering

The screenshot shows a web-based application for managing class schedules. At the top, there is a header bar with the university's logo and name. Below the header, the main content area displays a list of classes and a form for adding new ones.

Header: Nguyen Duc Nghia

Class List: Nguyen Ly Ngan Ngur Lap Trinh

Actions: Buoi hoc | Sinh vien | Yeu cau buoi hoc | Dong y | Tu chon

Class Details: So luong yeu cau: 20 | Thoi gian: 7:00 - 9:00 | Ngay: 10/11/2025

Add Class Form:

Môn học	Hình thức	Ngày	
Nguyen ly Ngan Ngur Lap Trinh	Trực tuyến	10/11/2025	
Tên buổi học	Địa điểm	Bắt đầu	Kết thúc
Tên buổi học...	Địa điểm	7:00	7:00

Buttons: Xác nhận



7.3 Phòng Đào tạo

The image displays three screenshots of the BK Tutor application interface:

- Dashboard:** Shows a large, empty light gray area, likely a placeholder for dynamic content.
- BK Tutor Trang chủ:** Shows a blank white page with the BK Tutor logo and "Trang chủ".
- Report View:** Shows a detailed report titled "TỔNG QUAN NGUỒN LỰC TRƯỜNG". It includes:
 - Filters: "Học Kỳ" (Search: HK2021) and "Mùa học" (Search: 2020-2021).
 - Key statistics:
 - Tổng số tutor: 250
 - Tổng số sinh viên: 3500
 - Tỷ lệ SV/TUTOR (TB): 14.0
 - A chart titled "PHÂN BỐ NGUỒN LỰC THEO KHOA" showing distribution across various faculties.
 - A section titled "Đánh giá chất lượng chương trình" with a table comparing four programs based on participation rate and average rating.



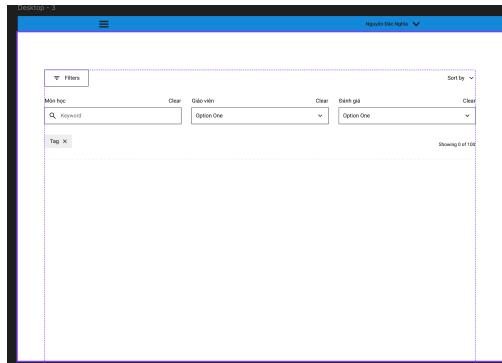
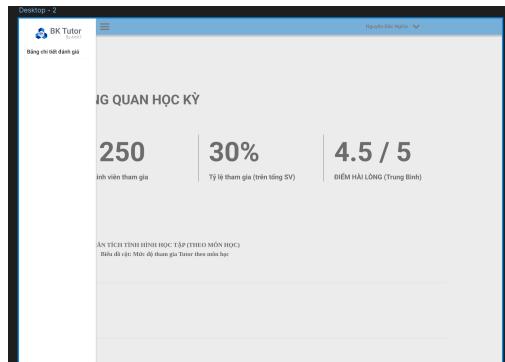
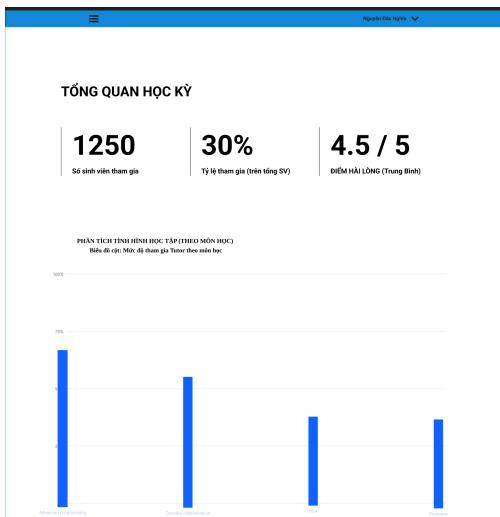
7.4 Phòng CTSV

The image displays six screenshots of the BK Tutor application interface, arranged in a 3x2 grid. Each screenshot shows a different page related to student management:

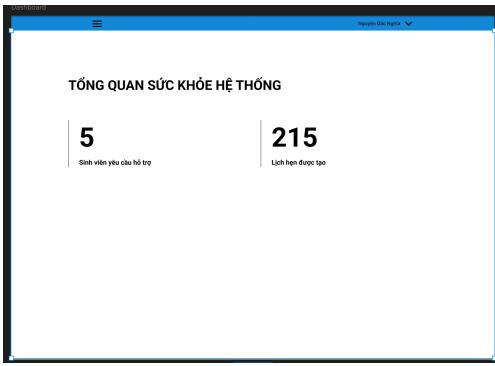
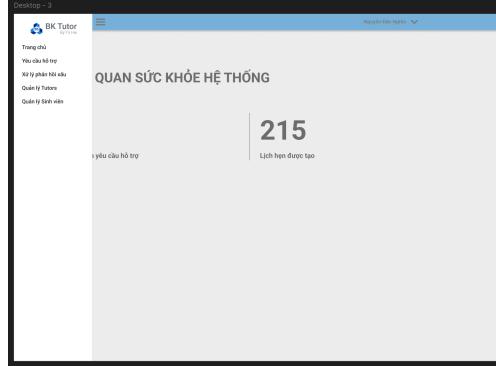
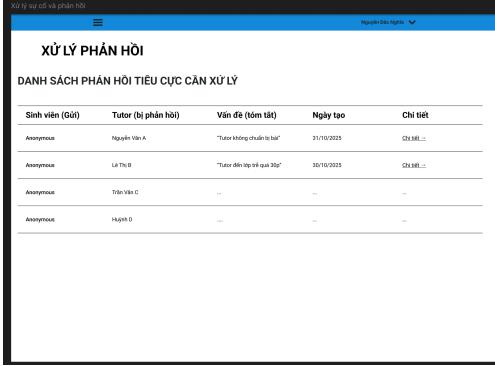
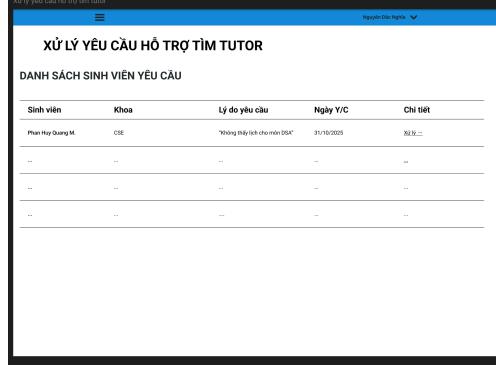
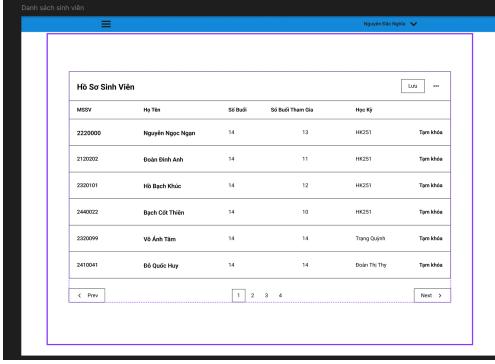
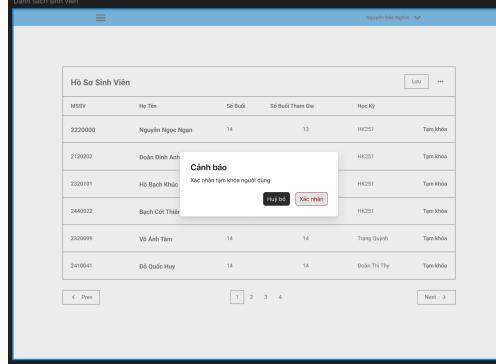
- Top Left:** A blank page with a blue header bar.
- Top Right:** A blank page with a blue header bar.
- Middle Left:** A search results page titled "Hồ Sơ Sinh Viên". It includes filters for "MSSV", "Họ Tên", "SD Buổi", "SD Buổi Tham Gia", and "Học Kỳ". A dropdown menu shows "HK231". Buttons for "Lưu" and "Xuất báo cáo" are at the bottom right.
- Middle Right:** A search results page titled "Nguyễn Ngọc Ngan". It lists student evaluations with columns for "Giảng viên đánh giá", "Điểm", "Môn", and "Ngày đánh giá". Each row includes a "Chi tiết" button. A dropdown menu shows "HK231". Buttons for "Lưu" and "Xuất báo cáo" are at the bottom right.
- Bottom Left:** A search results page titled "Danh Sách Buổi Tư Vấn". It includes filters for "Giảng Viên", "SD Buổi", "Môn Học", and "Học Kỳ". A dropdown menu shows "HK231". Buttons for "Lưu" and "Xuất báo cáo" are at the bottom right.
- Bottom Right:** A search results page titled "Đoàn Thị Thy". It lists student evaluations with columns for "Sinh Viên", "MSSV", "Điểm", and "CN NH". Each row includes a "Chi tiết" button. A dropdown menu shows "HK231". Buttons for "Lưu" and "Xuất báo cáo" are at the bottom right.



7.5 Khoa - Bộ môn



7.6 Điều phối viên

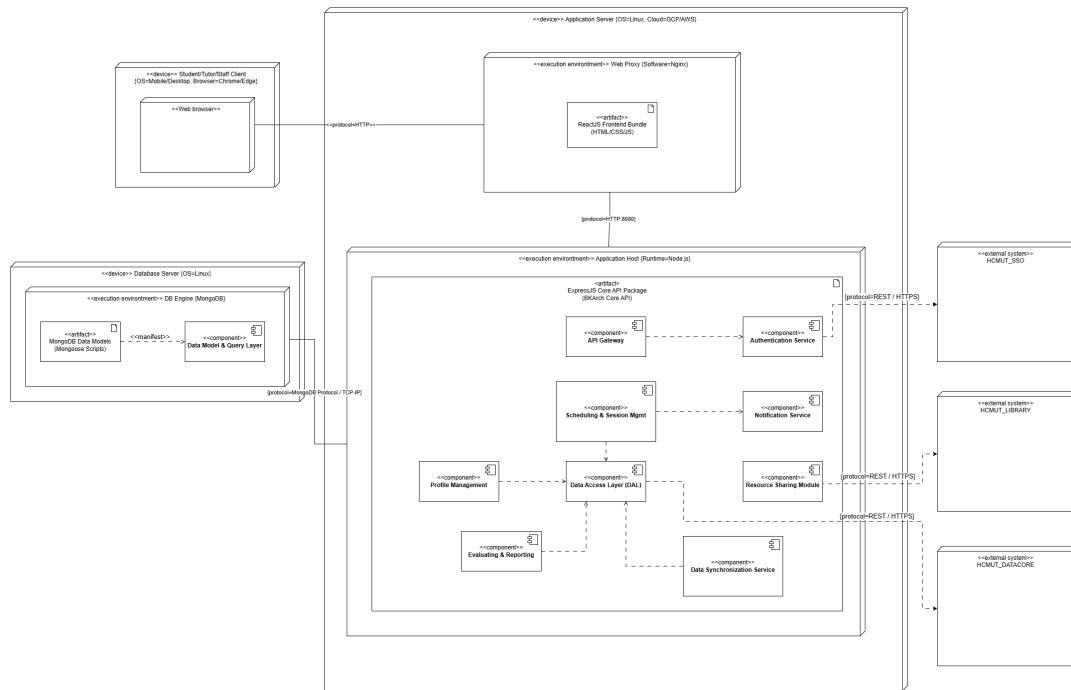


University of Technology, Ho Chi Minh City
Faculty of Computer Science and Engineering

Danh sách tutor					
MSTT	Họ Tên	Số Buổi Đăng ký	Số Buổi Dạy	Học Kỳ	Action
0001	Nguyễn Văn A	14	13	HK251	Tạm khóa
0002	Nguyễn Văn B	14	11	HK251	Tạm khóa
0003	Nguyễn Văn C	14	12	HK251	Tạm khóa
0004	Nguyễn Văn D	14	10	HK251	Tạm khóa
0005	Nguyễn Văn E	14	14	HK251	Tạm khóa
0006	Nguyễn Văn F	14	14	HK251	Tạm khóa

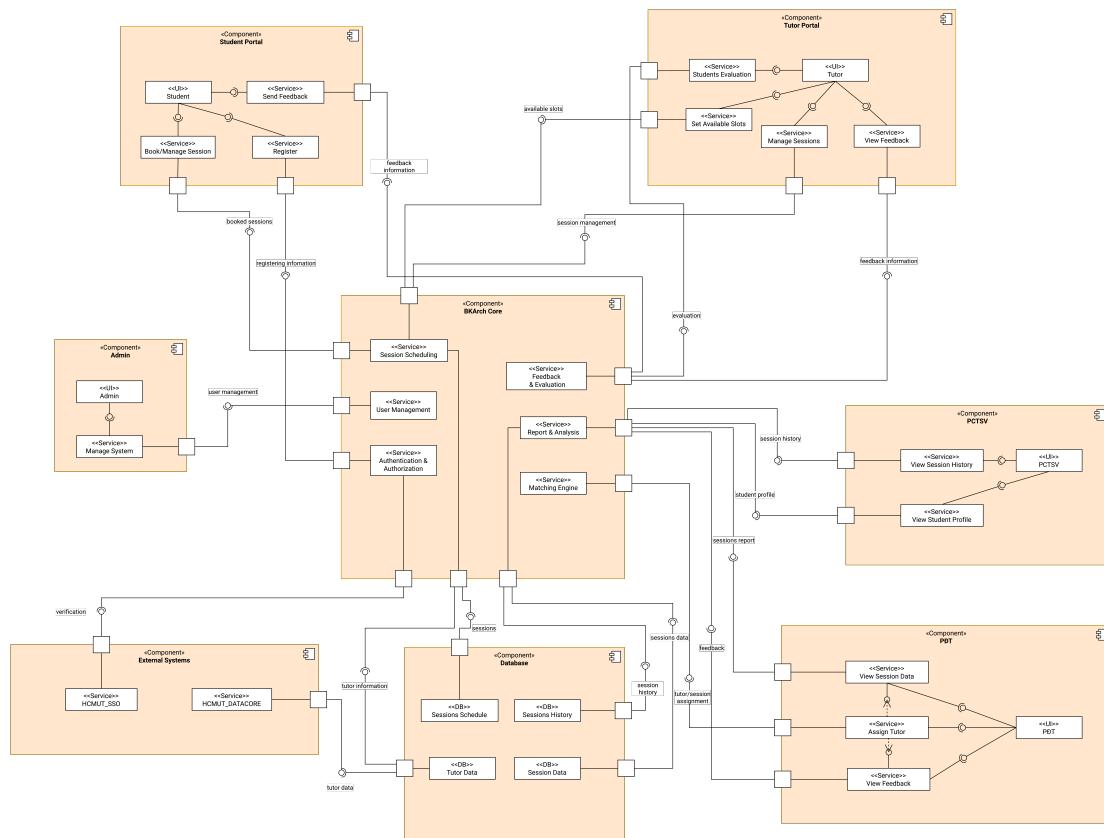
8 Các lược đồ cấu trúc (Structure Diagram)

8.1 Lược đồ triển khai (Deployment Diagram)



8.2 Lược đồ hiện thực (Implementation Diagram)

8.2.1 Lược đồ hiện thực các component



Mô tả các component:

Component diagram cung cấp cái nhìn tổng quan về kiến trúc module hóa của hệ thống, được xây dựng dựa trên thành phần trung tâm là BKArch Core. Thành phần này là trung tâm của hệ thống, chịu trách nhiệm chính về các dịch vụ cốt lõi như Session Scheduling (lên lịch buổi tư vấn), User Management (quản lý người dùng), Authentication & Authorization (xác thực và ủy quyền), cùng với Matching Engine (cơ chế ghép cặp Tutor Mentee) quan trọng.

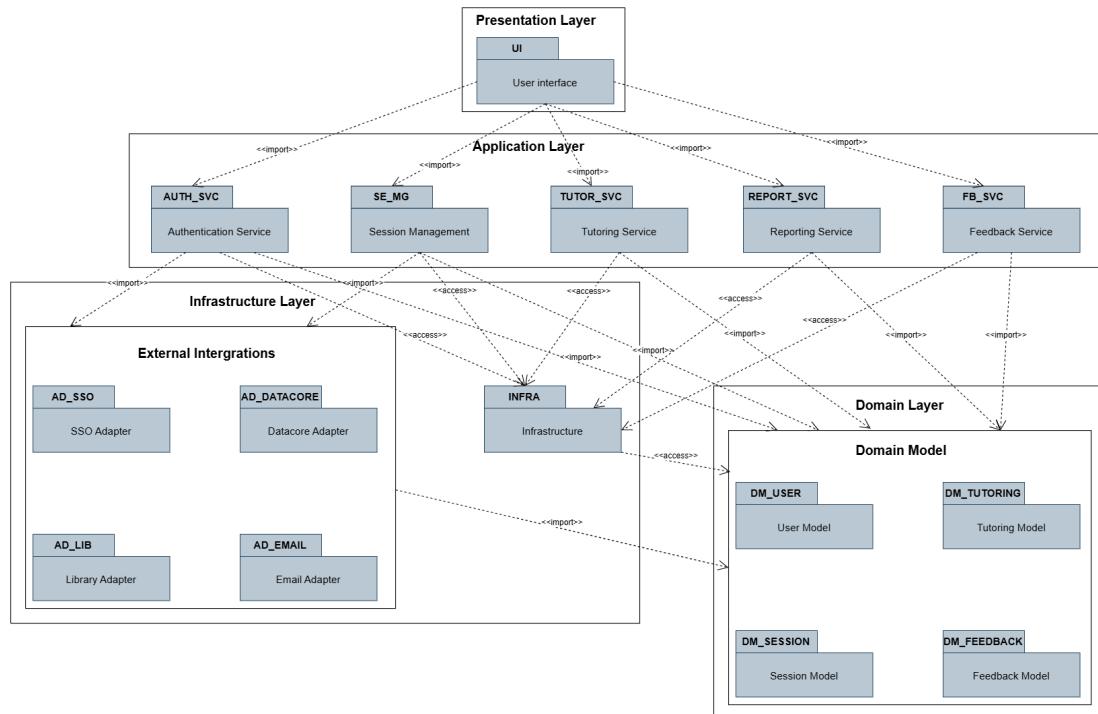
Các thành phần giao diện người dùng bao gồm Student Portal và Tutor Portal, phục vụ vai trò cụ thể: Student Portal cho phép sinh viên Đăng ký và Quản lý/Dặt Lịch, trong khi Tutor Portal tập trung vào việc Quản lý Buổi Tư Vấn, Dánh giá Sinh viên và Xem Phản hồi. Cả hai Portal đều sử dụng dịch vụ Evaluation do BKArch Core cung cấp và trao đổi dữ liệu Phiên với lõi hệ thống.

Về mặt dữ liệu, Components Database là kho lưu trữ tập trung, cung cấp các giao diện truy cập dữ liệu (DB) chuyên biệt cho Tutor Data, Session Data, Sessions History và Sessions Schedule. BKArch Core là thành phần tiêu thụ chính các dịch vụ từ Database.

Cuối cùng, hệ thống tích hợp các thành phần quản trị và ngoại vi. Admin quản lý người dùng và hệ thống thông qua BKArch Core. PDT (Phòng Đào Tạo) có trách nhiệm Assign Tutor (phân công) và View Session Data/Feedback. Hệ thống cũng thể hiện sự phụ thuộc vào External Systems như HCMUT-SSO và HCMUT-Datacore để lấy dữ liệu đầu vào cần thiết cho hoạt động. Mỗi quan hệ giữa các thành phần được định nghĩa rõ ràng qua các giao diện cung cấp (Provided

Interface, như các dịch vụ «Service») và các giao diện yêu cầu (Required Interface), đảm bảo luồng thông tin và nghiệp vụ diễn ra mạch lạc và có tổ chức.

8.2.2 Lược đồ hiện thực các Package

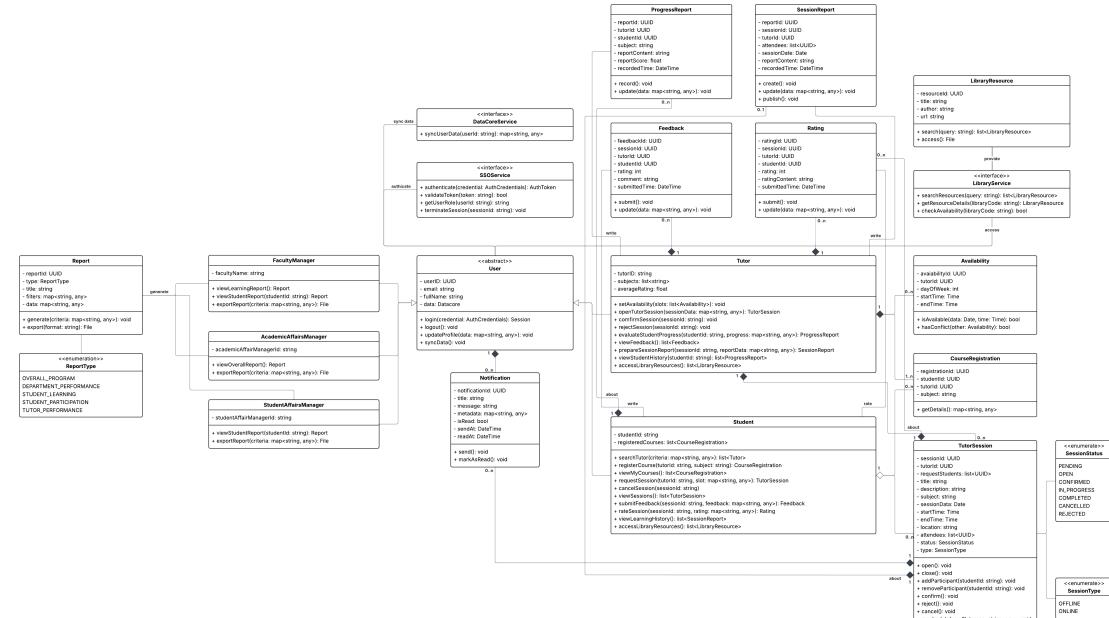


Mô tả các package:

- Presentation Layer gồm User Interface, chịu trách nhiệm nhận thao tác của người dùng và gửi các yêu cầu xuống Application Layer. UI chỉ sử dụng các dịch vụ qua quan hệ import và không xử lý nghiệp vụ nội bộ.
- Application Layer bao gồm năm module chính: Authentication Service, Tutoring Service, Session Management, Feedback Service và Reporting Service. Các module này thực thi nghiệp vụ của hệ thống và sử dụng Domain Model thông qua các quan hệ import. Đồng thời, các module truy cập Infrastructure qua access để đọc và ghi dữ liệu vào cơ sở dữ liệu. Một số module như Authentication Service và Session Management còn sử dụng External Integrations qua quan hệ import để liên kết với các hệ thống bên ngoài.
- Authentication Service sử dụng SSO Adapter và Datacore Adapter để xác thực người dùng và đồng bộ thông tin tài khoản từ hệ thống trường. Module này truy cập Infrastructure để lưu trữ và cập nhật dữ liệu người dùng.
- Tutoring Service quản lý các chức năng liên quan đến chương trình tutor–mentee như đăng ký hoặc ghép cặp. Module này sử dụng các mô hình nghiệp vụ từ Domain Model và truy cập Infrastructure để xử lý lưu trữ.

- Session Management chịu trách nhiệm quản lý lịch rảnh, tạo hoặc hủy buổi tư vấn và gửi thông báo. Module này sử dụng Email Adapter để gửi email và truy cập Infrastructure để lưu thông tin phiên làm việc.
 - Feedback Service cho phép sinh viên gửi đánh giá và tutor ghi nhận tiền bộ. Module sử dụng Domain Model để xử lý thông tin đánh giá và truy cập Infrastructure để lưu trữ phản hồi.
 - Reporting Service tổng hợp dữ liệu từ các module khác để tạo báo cáo cho Bộ môn, Phòng Đào tạo và P.CTSV. Module này sử dụng các mô hình domain liên quan và truy cập Infrastructure để thực hiện các truy vấn.
 - Domain Layer chứa các mô hình nghiệp vụ cốt lõi của hệ thống như User Model, Tutoring Model, Session Model và Feedback Model. Domain Model được Application Layer sử dụng qua quan hệ import. Infrastructure và External Integrations sử dụng Domain Model để ánh xạ và lưu trữ dữ liệu.
 - Infrastructure Layer cung cấp hạ tầng như repository, kết nối cơ sở dữ liệu, logging và configuration. Module Infrastructure truy cập Domain Model khi thao tác với dữ liệu nghiệp vụ.
 - External Integrations bao gồm SSO Adapter, Datacore Adapter, Library Adapter và Email Adapter. Các adapter này sử dụng Domain Model để chuyển đổi dữ liệu giữa hệ thống BKArch và các hệ thống bên ngoài.

8.3 Lược đồ các lớp hiện thực



Mô tả các lớp hiện thực:



Lớp: User (Abstract)

- **login(AuthCredentials credentials) : Session**

Xác thực người dùng thông qua HCMUT_SSO và tạo phiên đăng nhập mới. Phương thức này kiểm tra thông tin đăng nhập và trả về session token nếu thành công.

- **logout() : void**

Kết thúc phiên làm việc hiện tại của người dùng. Hủy session token và ghi lại thời điểm đăng xuất vào hệ thống.

- **updateProfile(Map<String,Any> data) : void**

Cập nhật thông tin cá nhân của người dùng như email, số điện thoại, tên đầy đủ. Phương thức này validate dữ liệu trước khi lưu vào database.

- **syncDataFromDataCore() : void**

Đồng bộ dữ liệu người dùng từ HCMUT_DATACORE. Cập nhật các thông tin như khoa, ngành, năm học từ hệ thống quản lý trung tâm.

Lớp: Student

- **searchTutor(Map<String,Any> criteria) : List<Tutor>**

Tìm kiếm Tutor dựa trên tiêu chí như môn học, khoa, loại Tutor, hình thức dạy học. Trả về danh sách các Tutor phù hợp đang chấp nhận sinh viên mới.

- **registerCourse(String tutorId, String subject) : CourseRegistration**

Đăng ký môn học với Tutor cụ thể (Instant Registration). Tạo CourseRegistration mới và gửi thông báo đến Tutor về sinh viên mới.

- **viewMyCourses() : List<CourseRegistration>**

Xem danh sách các khóa học (Tutor + Môn học) mà sinh viên đã đăng ký. Hiển thị thông tin về Tutor, môn học và trạng thái.

- **requestSession(String tutorId, Map<String,Any> slot) : TutorSession**

Gửi yêu cầu đặt lịch học với Tutor cho một thời gian cụ thể. Tutor sẽ nhận được thông báo và có thể xác nhận hoặc từ chối yêu cầu này.

- **cancelSession(String sessionId) : void**

Hủy buổi học đã đặt trước. Hệ thống sẽ gửi thông báo đến Tutor và cập nhật trạng thái session thành CANCELLED.

- **viewSessions() : List<TutorSession>**

Xem danh sách tất cả các buổi tư vấn/lịch hẹn của sinh viên. Bao gồm các buổi đã đăng ký tham gia, đã đặt lịch 1-1 với trạng thái khác nhau (PENDING, CONFIRMED, COMPLETED...).

- **submitFeedback(String sessionId, Map<String,Any> feedback) : Feedback**

Gửi phản hồi về buổi học bao gồm comment, điểm mạnh và điểm cần cải thiện của Tutor. Sinh viên có thể chọn gửi phản hồi ẩn danh hoặc công khai.

- **rateSession(String sessionId, Map<String,Any> rating) : Rating**

Dánh giá chi tiết buổi học theo các tiêu chí: chất lượng giảng dạy, kỹ năng giao tiếp, sự chuẩn bị và mức độ hữu ích. Điểm đánh giá sẽ ảnh hưởng đến averageRating của Tutor.



- **viewLearningHistory() : List<SessionReport>**

Theo dõi lịch sử học tập bao gồm các buổi học đã tham gia, nội dung đã học và đánh giá của Tutor. Giúp sinh viên xem lại quá trình học tập và tiến bộ của mình.

- **accessLibraryResources() : List<LibraryResource>**

Truy cập danh sách tài liệu từ thư viện HCMUT liên quan đến môn học đang theo học. Hệ thống tích hợp với HCMUT_LIBRARY để cung cấp tài liệu phù hợp.

Lớp: Tutor

- **setAvailability(List<Availability> slots) : void**

Thiết lập lịch rảnh cho việc dạy học bao gồm ngày, giờ, địa điểm.

- **openTutorSession(Map<String,Any> sessionData) : TutorSession**

Mở buổi tư vấn nhóm công khai cho sinh viên tham gia. Tutor cung cấp thông tin về chủ đề, thời gian, địa điểm.

- **confirmSession(String sessionId) : void**

Xác nhận yêu cầu đặt lịch hẹn từ sinh viên. Hệ thống sẽ gửi thông báo xác nhận đến sinh viên và cập nhật trạng thái session thành CONFIRMED.

- **rejectSession(String sessionId, String reason) : void**

Từ chối yêu cầu đặt lịch hẹn từ sinh viên với lý do cụ thể. Sinh viên sẽ nhận được thông báo.

- **viewSessions() : List<TutorSession>**

Xem danh sách tất cả các buổi tư vấn và lịch hẹn mà Tutor quản lý. Bao gồm các buổi đã mở, yêu cầu đặt lịch đang chờ xử lý và các buổi sắp diễn ra.

- **manageSession(String sessionId) : void**

Quản lý buổi học bao gồm cập nhật thông tin, thêm/xóa sinh viên, thay đổi trạng thái session. Tutor có toàn quyền kiểm soát các buổi học do mình tạo.

- **evaluateStudentProgress(String studentId, Map<String,Any> progress) : ProgressReport**

Dánh giá tiến bộ học tập của sinh viên theo từng môn học.

- **viewFeedback() : List<Feedback>**

Xem tất cả phản hồi từ sinh viên về các buổi học. Giúp Tutor nắm bắt ý kiến đánh giá và cải thiện chất lượng giảng dạy.

- **prepareSessionReport(String sessionId) : SessionReport**

Tổng hợp biên bản buổi học bao gồm nội dung đã dạy, mục tiêu đạt được, đánh giá sinh viên và khuyến nghị. Biên bản này được lưu vào hệ thống để theo dõi lịch sử.

- **viewStudentHistory(String studentId) : List<SessionReport>**

Xem lịch sử học tập của một sinh viên cụ thể bao gồm các buổi học đã tham gia và biên bản các buổi học. Giúp Tutor hiểu rõ hơn về quá trình học tập của sinh viên.



Lớp: FacultyManager

- **viewLearningReport() : Report**

Xem báo cáo về kết quả học tập của sinh viên trong khoa thông qua chương trình Tutor. Phân tích mức độ cải thiện học tập và hiệu quả của chương trình hỗ trợ.

- + **viewStudentReport(studentId: string): Report**

Phân tích tiến bộ học tập của sinh viên khoa qua các chỉ số như GPA trước/sau tham gia, số môn đã cải thiện, mức độ tham gia. Cung cấp insights để điều chỉnh chính sách hỗ trợ.

- **exportReport(Map<String,Any> criteria) : File**

Xuất báo cáo phân tích kết quả học tập ra file. Hỗ trợ việc báo cáo cho Ban Giám hiệu về hiệu quả chương trình.

Lớp: AcademicAffairsManager

- **viewOverallReport() : Report**

Xem báo cáo tổng thể về chương trình Tutor trên toàn trường. Bao gồm dữ liệu từ tất cả các khoa, phân tích xu hướng và đề xuất cải thiện.

- **exportReport(Map<String,Any> criteria) : File**

Xuất báo cáo tổng thể ra file để trình bày cho BGH. Hỗ trợ việc ra quyết định về chính sách và ngân sách cho chương trình.

Lớp: StudentAffairsManager

- **viewStudentReport(studentId: string): Report**

Phân tích tiến bộ học tập của sinh viên khoa qua các chỉ số như GPA trước/sau tham gia, số môn đã cải thiện, mức độ tham gia. Cung cấp insights để điều chỉnh chính sách hỗ trợ.

- **exportReport(Map<String,Any> criteria) : File**

Xuất báo cáo về hoạt động và điểm rèn luyện của sinh viên ra file. Hỗ trợ công tác quản lý và đánh giá sinh viên.

Lớp: Availability

- **isAvailable(Date date, Time time) : Boolean**

Kiểm tra xem Tutor có rảnh vào ngày giờ cụ thể hay không.

- **hasConflict(Availability otherSlot) : Boolean**

Kiểm tra xung đột thời gian giữa hai khung giờ rảnh. Đảm bảo Tutor không thiết lập lịch rảnh trùng lặp hoặc chồng chéo.

Lớp: TutorSession

- **open() : void**

Mở buổi tư vấn để sinh viên có thể đăng ký tham gia. Cập nhật trạng thái session thành OPEN và gửi thông báo đến sinh viên phù hợp.

- **close() : void**

Đóng buổi tư vấn, không cho phép thêm sinh viên mới. Có thể đóng khi đã đủ số lượng hoặc khi sắp đến giờ học.



- **addParticipant(String studentId) : void**

Thêm sinh viên vào danh sách tham gia buổi tư vấn. Kiểm tra số lượng tối đa trước khi cho phép tham gia.

- **removeParticipant(String studentId) : void**

Xóa sinh viên khỏi danh sách tham gia buổi tư vấn. Sinh viên có thể tự hủy hoặc Tutor có thể xóa khi cần thiết.

- **confirm() : void**

Xác nhận buổi học/lịch hẹn. Cập nhật trạng thái thành CONFIRMED và gửi thông báo xác nhận đến tất cả người tham gia.

- **reject(String reason) : void**

Từ chối yêu cầu đặt lịch hoặc hủy buổi tư vấn với lý do cụ thể. Gửi thông báo từ chối kèm lý do đến người liên quan.

- **cancel() : void**

Hủy buổi học đã xác nhận. Gửi thông báo hủy đến tất cả sinh viên đã đăng ký và cập nhật trạng thái thành CANCELLED.

- **reschedule(Map<String,Any> newSlot) : void**

Đổi lịch buổi học sang thời gian mới. Gửi thông báo đổi lịch đến tất cả người tham gia và yêu cầu xác nhận lại.

Lớp: CourseRegistration

- **getDetails() : Map<String,Any>**

Lấy thông tin chi tiết về đăng ký bao gồm thông tin Student, Tutor và Subject được populate đầy đủ. Trả về dữ liệu hoàn chỉnh để hiển thị.

Lớp: SessionReport

- **create() : void**

Tạo biên bản mới cho buổi học. Tutor nhập thông tin về nội dung, mục tiêu, kết quả đạt được và đánh giá sinh viên.

- **update(Map<String,Any> data) : void**

Cập nhật thông tin biên bản buổi học. Cho phép Tutor sửa đổi nội dung trước khi publish hoặc sau khi nhận phản hồi.

- **publish() : void**

Công bố biên bản buổi học cho sinh viên xem. Sau khi publish, sinh viên có thể truy cập và xem nội dung biên bản.

Lớp: ProgressReport

- **record() : void**

Ghi nhận báo cáo tiến bộ của sinh viên. Tutor nhập thông tin về kỹ năng đã cải thiện, điểm cần khắc phục và điểm số đánh giá.

- **update(Map<String,Any> data) : void**

Cập nhật báo cáo tiến bộ đã ghi nhận. Cho phép Tutor điều chỉnh đánh giá khi có thêm thông tin mới.



Lớp: LibraryResource

- **search(String query) : List<LibraryResource>**

Tìm kiếm tài liệu trong thư viện theo từ khóa. Tích hợp với HCMUT_LIBRARY để truy vấn và trả về danh sách tài liệu phù hợp.

- **access() : File**

Truy cập file tài liệu từ thư viện. Phương thức này xử lý việc download hoặc xem online tài liệu từ hệ thống thư viện.

Lớp: Feedback

- **submit() : void**

Gửi phản hồi về Tutor sau buổi học.

- **update(Map<String,Any> data) : void**

Cập nhật phản hồi đã gửi. Cho phép sinh viên chỉnh sửa nội dung phản hồi trong thời gian cho phép.

Lớp: Rating

- **submit() : void**

Gửi rating về Tutor sau buổi học.

- **update(Map<String,Float> scores) : void**

Cập nhật điểm đánh giá theo từng tiêu chí. Cho phép sinh viên điều chỉnh đánh giá và tự động tính lại điểm overall.

Lớp: Notification

- **send() : void**

Gửi thông báo đến người dùng.

- **markAsRead() : void**

Dánh dấu thông báo đã đọc. Cập nhật trạng thái isRead thành true và ghi nhận thời gian đọc.

Lớp: Report

- **generate(Map<String,Any> criteria) : void**

Tạo báo cáo dựa trên tiêu chí lọc. Truy vấn dữ liệu từ database, phân tích và tổng hợp thành báo cáo theo định dạng yêu cầu.

- **export(String format) : File**

Xuất báo cáo ra file với định dạng chỉ định (PDF, Excel, CSV). Hỗ trợ việc lưu trữ và chia sẻ báo cáo.

Lớp: SSOService (Interface)

- **authenticate(AuthCredentials credentials) : AuthToken**

Xác thực thông tin đăng nhập của người dùng thông qua HCMUT_SSO. Trả về auth token nếu thông tin đăng nhập hợp lệ.



- **validateToken(String token) : Boolean**

Kiểm tra tính hợp lệ của token. Xác minh token chưa hết hạn và chưa bị thu hồi.

- **getUserRole(String userId) : String**

Lấy vai trò của người dùng từ hệ thống SSO. Xác định quyền truy cập và chức năng mà người dùng được phép sử dụng.

- **terminateSession(String sessionId) : void**

Kết thúc phiên đăng nhập. Vô hiệu hóa session token và logout người dùng khỏi hệ thống.

Lớp: DataCoreService (Interface)

- **syncUserData(String userId) : Map<String, Any>**

Dòng bộ dữ liệu người dùng từ HCMUT_DATACORE. Lấy thông tin mới nhất về sinh viên/nhân viên và cập nhật vào hệ thống.

Lớp: LibraryService (Interface)

- **searchResources(String query) : List<LibraryResource>**

Tìm kiếm tài liệu trong thư viện HCMUT theo từ khóa. Trả về danh sách tài liệu phù hợp từ hệ thống HCMUT_LIBRARY.

- **getResourceDetails(String resourceId) : LibraryResource**

Lấy thông tin chi tiết của tài liệu từ thư viện. Bao gồm tác giả, năm xuất bản, mã tài liệu và URL truy cập.

- **checkAvailability(String resourceId) : Boolean**

Kiểm tra tình trạng sẵn có của tài liệu trong thư viện. Xác định tài liệu có đang được mượn hay có sẵn để truy cập.