

ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH



BÁO CÁO BÀI TẬP LỚN

SEAWHISPER

ỨNG DỤNG CHIA SẺ ẨN DANH & CHỮA LÀNH TÂM HỒN

Môn học: Phát triển Ứng dụng Di động

Học kỳ: 2025–2026 (HK251)

Giảng viên: Hoàng Lê Hải Thanh

Sinh viên thực hiện **MSSV**

Trần Minh Khang 2211472

Hà Thái Toàn 2213524

Đoàn Nhật Tiến 22

Lê Minh Tuấn 22

TP. Hồ Chí Minh, năm 2026

Tóm tắt

SeaWhisper là ứng dụng di động chia sẻ ẩn danh theo mô hình “thả chai” (message in a bottle), nơi người dùng có thể gửi những tâm sự của mình ra “biển” và một người khác sẽ ngẫu nhiên “nhặt” được để đọc và phản hồi một cách nhẹ nhàng. Dự án hướng đến trải nghiệm chữa lành (healing), giảm độc hại (toxic) và khuyến khích tương tác tích cực thông qua cơ chế ẩn danh có kiểm duyệt.

Mục tiêu của hệ thống là tạo ra một không gian an toàn, không phán xét để người dùng giải tỏa cảm xúc; đồng thời áp dụng bộ lọc nội dung (AI Filter) nhằm hạn chế ngôn từ công kích, kích động, hoặc tiêu cực quá mức. Về mặt kỹ thuật, ứng dụng được xây dựng với React Native cho phía client, FastAPI cho phía server, Redis làm cơ sở dữ liệu lưu trữ nhanh và cơ chế ghép ngẫu nhiên (matching) để phân phối “chai” giữa người gửi và người nhận.

Báo cáo trình bày quy trình phát triển theo hướng lặp (iterative), các quyết định thiết kế MVP, kiến trúc hệ thống, thiết kế web service, chiến lược triển khai, kiểm thử và đánh giá trải nghiệm người dùng. Kết quả khảo sát ban đầu cho thấy nhu cầu cao đối với một nền tảng ẩn danh lành mạnh; các phân tích hành vi và phản hồi người dùng được sử dụng để đề xuất cải tiến cho các phiên bản tiếp theo, bao gồm tối ưu bộ lọc, đa dạng hóa cơ chế phản hồi cảm xúc và tăng cường tính an toàn cộng đồng.

Mục lục

Tóm tắt	1
1 Giới thiệu	3
1.1 Bối cảnh và vấn đề	3
1.2 Mục tiêu dự án	3
1.3 Phạm vi và đối tượng hưởng tới	3
1.4 Giá trị thực tiễn và tác động	3
1.5 Cấu trúc báo cáo	4
2 Phương pháp và quy trình thực hiện	4
2.1 Cách tiếp cận phát triển	4
2.2 Quy trình làm việc nhóm	4
2.3 Công cụ và nền tảng sử dụng	5
2.4 Quyết định phạm vi MVP	5
2.5 Đánh giá phương pháp	5
3 Tính năng và chức năng hệ thống	5
3.1 Tổng quan các tính năng	5
3.2 Gửi chai (Send Bottle)	6
3.3 Nhận chai ngẫu nhiên (Receive Bottle)	6
3.4 Phản hồi cảm xúc	6
3.5 Lưu trữ và xem lại chai	7
3.6 Giới hạn của MVP	7
4 Triển khai hệ thống	7
4.1 Tổng quan	7
4.2 Triển khai Frontend với React Native Expo	7
4.3 Quản lý trạng thái và điều hướng	8
4.4 Triển khai Backend	8
4.5 Cơ sở dữ liệu Redis	9
4.6 Cơ chế phân phối ngẫu nhiên	9
4.7 Giới hạn triển khai	9

1 Giới thiệu

1.1 Bối cảnh và vấn đề

Trong bối cảnh mạng xã hội phát triển mạnh, nhu cầu chia sẻ cảm xúc cá nhân của người trẻ ngày càng tăng. Tuy nhiên, việc công khai danh tính thường đi kèm rủi ro bị phán xét, bị soi xét đời tư hoặc gặp các phản hồi tiêu cực. Các ứng dụng ẩn danh hiện có giúp giảm rào cản chia sẻ, nhưng lại dễ trở thành môi trường độc hại do thiếu cơ chế kiểm duyệt và định hướng tương tác tích cực. Điều này khiến người dùng có thể bị ảnh hưởng tâm lý sau khi tiếp nhận các nội dung hoặc bình luận tiêu cực.

1.2 Mục tiêu dự án

Dự án SeaWhisper được xây dựng nhằm tạo ra một không gian ẩn danh nhưng an toàn, hướng đến trải nghiệm chữa lành. Các mục tiêu chính gồm:

- Cung cấp cơ chế chia sẻ ẩn danh theo mô hình “thả chai” và “nhặt chai”, giúp người dùng giải bày cảm xúc một cách tự nhiên.
- Khuyến khích tương tác đồng cảm, hạn chế tranh cãi và công kích bằng cách ưu tiên phản hồi nhẹ nhàng.
- Áp dụng bộ lọc nội dung (AI Filter) để giảm toxic, hạn chế nội dung gây hại và nâng cao mức độ an toàn cộng đồng.
- Xây dựng trải nghiệm sử dụng thư giãn, tối giản và mang cảm giác “ocean-inspired”.

1.3 Phạm vi và đối tượng hướng tới

SeaWhisper tập trung vào nhóm người dùng trẻ (đặc biệt Gen Z), sinh viên và những người hướng nội có nhu cầu giải tỏa tâm trạng, tìm kiếm sự đồng cảm nhưng không muốn lộ danh tính. Trong phạm vi MVP, dự án ưu tiên các chức năng cốt lõi gồm: gửi “chai”, nhận ngẫu nhiên “chai” và phản hồi cảm xúc đơn giản.

1.4 Giá trị thực tiễn và tác động

SeaWhisper hướng tới mô hình “C2C cảm xúc”, nơi mỗi người vừa là người chia sẻ vừa là người lắng nghe. Nếu triển khai thực tế, ứng dụng có thể:

- Giảm áp lực tâm lý khi người dùng có kênh giải bày an toàn.
- Hạn chế sự lan truyền độc hại thường thấy trong các nền tảng ẩn danh.
- Khuyến khích hành vi đồng cảm và hỗ trợ tinh thần giữa người dùng.

1.5 Cấu trúc báo cáo

Phần còn lại của báo cáo được tổ chức như sau:

- Phần **Phương pháp và quy trình phát triển**: mô tả cách nhóm tổ chức công việc, công cụ cộng tác và lý do lựa chọn giải pháp.
- Phần **Tính năng và chức năng**: trình bày MVP và các mở rộng.
- Phần **Triển khai**: mô tả công nghệ, kiến trúc và thiết kế API.
- Phần **Đánh giá**: kiểm thử, khảo sát người dùng, phân tích hành vi và đề xuất cải tiến.

2 Phương pháp và quy trình thực hiện

2.1 Cách tiếp cận phát triển

Dự án SeaWhisper được phát triển theo hướng tiếp cận lặp (iterative development), tập trung vào việc xây dựng và hoàn thiện các chức năng cốt lõi của MVP trước khi mở rộng thêm các tính năng nâng cao. Nhóm ưu tiên triển khai những thành phần mang lại giá trị trực tiếp cho người dùng trong thời gian giới hạn của học kỳ.

Thay vì xây dựng một hệ thống phức tạp ngay từ đầu, dự án tập trung vào việc hiện thực hóa ý tưởng chính là mô hình “thả chai – nhặt chai”, đảm bảo luồng sử dụng đơn giản, ổn định và dễ mở rộng trong tương lai.

2.2 Quy trình làm việc nhóm

Nhóm áp dụng quy trình làm việc theo các bước sau:

- Phân tích yêu cầu và xác định phạm vi MVP.
- Thiết kế luồng người dùng (user flow) và kiến trúc tổng thể.
- Phân chia nhiệm vụ theo vai trò Frontend và Backend.
- Triển khai từng chức năng độc lập, sau đó tích hợp hệ thống.
- Kiểm thử và tinh chỉnh dựa trên trải nghiệm sử dụng.

Các nhiệm vụ và tiến độ được quản lý thông qua GitHub Project Board, cho phép theo dõi trạng thái công việc và phối hợp giữa các thành viên một cách hiệu quả.

2.3 Công cụ và nền tảng sử dụng

Trong quá trình phát triển, nhóm sử dụng các công cụ sau:

- **GitHub**: quản lý mã nguồn, phân chia nhánh và theo dõi tiến độ.
- **Figma**: thiết kế giao diện và mô phỏng luồng người dùng.
- **Visual Studio Code**: môi trường phát triển chính cho cả frontend và backend.
- **Postman**: kiểm thử và xác minh các API backend.

2.4 Quyết định phạm vi MVP

Do giới hạn về thời gian và nguồn lực, nhóm quyết định tập trung vào các chức năng thiết yếu nhất của hệ thống, bao gồm:

- Gửi nội dung ẩn danh dưới dạng “chai”.
- Nhận ngẫu nhiên “chai” từ người dùng khác.
- Phản hồi cảm xúc cơ bản và lưu trữ lịch sử tương tác.

Các chức năng nâng cao như phân tích cảm xúc tự động hoặc bộ lọc thông minh được xác định là hướng phát triển trong các phiên bản tiếp theo, nhưng không nằm trong phạm vi triển khai của MVP.

2.5 Đánh giá phương pháp

Cách tiếp cận tập trung vào MVP giúp nhóm:

- Hoàn thành được sản phẩm khả dụng trong thời gian học kỳ.
- Giảm độ phức tạp trong quá trình triển khai và kiểm thử.
- Dễ dàng mở rộng hệ thống trong tương lai khi có thêm thời gian và tài nguyên.

3 Tính năng và chức năng hệ thống

3.1 Tổng quan các tính năng

SeaWhisper được thiết kế xoay quanh trải nghiệm chia sẻ ẩn danh mang tính chữa lành. Trong phạm vi MVP, hệ thống tập trung vào các chức năng cốt lõi nhằm đảm bảo luồng sử dụng đơn giản, ổn định và dễ tiếp cận đối với người dùng lần đầu.

Các tính năng được ưu tiên triển khai bao gồm gửi “chai”, nhận “chai” ngẫu nhiên và phản hồi cảm xúc cơ bản. Những chức năng này phản ánh trực tiếp ý tưởng trung tâm của dự án và đủ để đánh giá tính khả thi của sản phẩm.

3.2 Gửi chai (Send Bottle)

Chức năng gửi chai cho phép người dùng viết và gửi một thông điệp ẩn danh ra “biển”. Nội dung được lưu trữ trên hệ thống mà không gắn với thông tin nhận dạng cá nhân của người gửi.

Quy trình gửi chai gồm các bước:

- Người dùng nhập nội dung tâm sự dưới dạng văn bản.
- Hệ thống tạo một đối tượng “chai” và lưu trữ vào cơ sở dữ liệu.
- Chai được đưa vào hàng đợi chờ người dùng khác nhặt.

Thiết kế này giúp giảm rào cản tâm lý khi chia sẻ và khuyến khích người dùng bộc lộ cảm xúc thật.

3.3 Nhận chai ngẫu nhiên (Receive Bottle)

Chức năng nhận chai cho phép người dùng ngẫu nhiên nhận được một thông điệp từ người khác. Việc phân phối chai không dựa trên mối quan hệ hay hồ sơ người dùng, đảm bảo tính ẩn danh và bất ngờ trong trải nghiệm.

Khi người dùng chọn nhặt chai:

- Hệ thống lấy một chai chưa được đọc từ cơ sở dữ liệu.
- Nội dung chai được hiển thị cho người dùng.
- Trạng thái chai được cập nhật để tránh phân phối trùng lặp.

Cơ chế này giúp tạo cảm giác kết nối ngẫu nhiên và khuyến khích hành vi lắng nghe.

3.4 Phản hồi cảm xúc

Thay vì cho phép bình luận trực tiếp, SeaWhisper trong MVP cung cấp cơ chế phản hồi cảm xúc đơn giản. Người nhận có thể gửi lại một phản hồi nhẹ nhàng nhằm thể hiện sự đồng cảm.

Các hình thức phản hồi trong MVP gồm:

- Biểu tượng cảm xúc (emoji) được định nghĩa sẵn.
- Tin nhắn ngắn mang tính động viên.

Cách tiếp cận này giúp hạn chế tranh cãi, giảm khả năng phát sinh nội dung tiêu cực và giữ đúng định hướng “healing” của ứng dụng.

3.5 Lưu trữ và xem lại chai

Hệ thống cho phép người dùng lưu lại các chai đã nhận hoặc đã gửi để xem lại sau. Tính năng này giúp người dùng ghi nhớ những khoảnh khắc có ý nghĩa và tăng mức độ gắn bó với ứng dụng.

Trong MVP, chức năng lưu trữ được triển khai ở mức cơ bản, chủ yếu phục vụ mục đích xem lại lịch sử tương tác.

3.6 Giới hạn của MVP

Do giới hạn về thời gian và phạm vi, MVP chưa triển khai các chức năng nâng cao như:

- Phân tích cảm xúc tự động.
- Bộ lọc nội dung thông minh.
- Cá nhân hóa trải nghiệm theo hành vi người dùng.

Những chức năng này được xác định là hướng mở rộng cho các phiên bản tiếp theo của SeaWhisper.

4 Triển khai hệ thống

4.1 Tổng quan

SeaWhisper được triển khai theo mô hình client-server, trong đó ứng dụng di động đóng vai trò là client và hệ thống backend chịu trách nhiệm xử lý nghiệp vụ, lưu trữ dữ liệu và phân phối nội dung giữa người dùng. Cách tiếp cận này giúp tách biệt rõ ràng giữa giao diện người dùng và logic xử lý, đồng thời tạo điều kiện thuận lợi cho việc mở rộng và bảo trì hệ thống.

Trong phạm vi MVP, hệ thống tập trung vào việc đảm bảo các chức năng cốt lõi hoạt động ổn định, với kiến trúc đơn giản nhưng đủ linh hoạt cho các phiên bản tiếp theo.

4.2 Triển khai Frontend với React Native Expo

Phía frontend của ứng dụng được phát triển bằng React Native kết hợp với Expo. Expo được lựa chọn nhằm đơn giản hóa quá trình thiết lập môi trường, giảm thời gian cấu hình ban đầu và hỗ trợ tốt cho việc phát triển nhanh ứng dụng di động đa nền tảng.

Các lợi ích chính khi sử dụng Expo trong dự án bao gồm:

- Không cần cấu hình thủ công Android SDK hoặc iOS build trong giai đoạn phát triển.
- Hỗ trợ hot reload, giúp tăng tốc quá trình thử nghiệm và chỉnh sửa giao diện.
- Cung cấp sẵn nhiều API phổ biến phục vụ phát triển ứng dụng di động.

Ứng dụng được tổ chức theo hướng component-based, trong đó mỗi màn hình (screen) và mỗi thành phần giao diện được tách thành các component độc lập. Điều này giúp mã nguồn dễ đọc, dễ tái sử dụng và thuận tiện cho việc mở rộng tính năng.

Các chức năng chính phía frontend bao gồm:

- Màn hình gửi chai: cho phép người dùng nhập nội dung tâm sự và gửi lên hệ thống.
- Màn hình nhật chai: hiển thị ngẫu nhiên một chai từ người dùng khác.
- Màn hình phản hồi: cho phép gửi phản hồi cảm xúc hoặc tin nhắn ngắn.
- Màn hình lưu trữ: hiển thị danh sách các chai đã gửi hoặc đã nhận.

Giao tiếp với backend được thực hiện thông qua các API REST, sử dụng giao thức HTTP và định dạng dữ liệu JSON.

4.3 Quản lý trạng thái và điều hướng

Ứng dụng sử dụng cơ chế quản lý trạng thái cục bộ để lưu trữ thông tin tạm thời như nội dung chai đang hiển thị hoặc lịch sử chai đã nhận trong phiên làm việc. Việc điều hướng giữa các màn hình được thiết kế theo luồng đơn giản, đảm bảo người dùng có thể thực hiện các thao tác chính chỉ với số bước tối thiểu.

Thiết kế điều hướng ưu tiên trải nghiệm liền mạch, hạn chế các thao tác phức tạp nhằm giữ đúng định hướng “healing” của ứng dụng.

4.4 Triển khai Backend

Backend của hệ thống được xây dựng bằng FastAPI, một framework Python hiện đại, phù hợp cho việc xây dựng các dịch vụ web hiệu năng cao. FastAPI được lựa chọn nhờ khả năng xử lý bất đồng bộ, cú pháp rõ ràng và hỗ trợ tự động sinh tài liệu API.

Các chức năng chính của backend bao gồm:

- Tiếp nhận và xử lý yêu cầu gửi chai từ client.
- Phân phối ngẫu nhiên chai cho người dùng khi có yêu cầu nhật chai.
- Quản lý trạng thái chai nhằm tránh việc một chai được phân phối nhiều lần.

Trong MVP, hệ thống không yêu cầu cơ chế xác thực phức tạp. Người dùng được định danh tạm thời bằng các mã định danh ngẫu nhiên do backend sinh ra, đảm bảo tính ẩn danh trong quá trình sử dụng.

4.5 Cơ sở dữ liệu Redis

Redis được sử dụng làm cơ sở dữ liệu chính trong phạm vi MVP. Với đặc tính lưu trữ dữ liệu trong bộ nhớ và hỗ trợ tốt cho các thao tác hàng đợi, Redis phù hợp cho việc lưu trữ và phân phối các “chai” trong SeaWhisper.

Mỗi chai được lưu trữ dưới dạng một đối tượng dữ liệu bao gồm nội dung, trạng thái và thời điểm tạo. Việc sử dụng Redis giúp:

- Giảm độ trễ khi truy xuất dữ liệu.
- Đơn giản hóa việc triển khai cơ chế lấy chai ngẫu nhiên.
- Phù hợp với dữ liệu mang tính tạm thời của MVP.

4.6 Cơ chế phân phối ngẫu nhiên

Khi người dùng thực hiện thao tác nhặt chai, backend sẽ truy xuất một chai chưa được đọc từ Redis theo cơ chế ngẫu nhiên. Sau khi chai được trả về cho client, trạng thái của chai được cập nhật để tránh việc phân phối lại cho người dùng khác.

Cơ chế này đảm bảo tính công bằng, ngẫu nhiên và mang lại trải nghiệm bất ngờ cho người dùng, đúng với ý tưởng ban đầu của dự án.

4.7 Giới hạn triển khai

Do tập trung vào hoàn thiện MVP trong thời gian giới hạn, hệ thống hiện tại chưa triển khai các chức năng nâng cao như lưu trữ lâu dài, phân tích nội dung tự động hoặc cá nhân hóa trải nghiệm người dùng. Tuy nhiên, kiến trúc được thiết kế theo hướng mở, cho phép tích hợp thêm các thành phần này trong các phiên bản tiếp theo.