1. Chỉ định phương pháp phân tách (decomposition) mà bạn đã sử dụng khi thiết kế sản phẩm phần mềm. Tại sao bạn lại chọn phương pháp phân hủy này?

Sử dụng **phương pháp thiết kế hướng đối tượng**. Đây là hướng tiếp cận khá phổ biến trong thiết kế phần mềm hiện nay.

- Cách tiếp cận hướng đối tượng:

+ tập trung vào cả hai khía cạnh của hệ thống là dữ liệu và hành động.

+ là một lối tư duy theo cách ánh xạ các thành phần trong bài bài toán vào các đối tượng ngoài đời thực.

* Với cách tiếp cận này, một hệ thống được chia tương ứng thành các phần nhỏ gọi là đối tượng. Mỗi đối tượng bao gồm đầy đủ cả dữ liệu và hành động liên quan đến đối tượng đó. Các đối tượng trong một hệ thống tương đối độc lập với nhau và phần mềm sẽ được xây dựng bằng cách kết hợp các đối tượng đó lại với nhau thông qua các mối quan hệ và tương tác giữa chúng.

- Lý do sử dụng cách tiếp cận này

+ Gần gũi với thế giới thực. Các thực thể như môn học, câu hỏi, bài test rất thân quen, gần gũi, ta dễ dàng trừu tượng hóa chúng thành các lớp, để từ mỗi lớp, ta tạo ra được đối tượng có các thuộc tính và các phương thức tương ứng với thực thể đó ngoài đời thực

+ Tính tái sử dụng

+ Đóng gói, che giấu thông tin -> hệ thống đáng tin cậy hơn

1. Bạn đã sử dụng những công nghệ (technology) nào khi phát triển sản phẩm phần mềm của mình?

Xây dựng phần mềm theo hướng Web API, tách rời back end và front end của trang web, giúp giảm sự phụ thuộc lẫn nhau của 2 phần.

- Database: Sử dụng cơ sở dữ liệu quan hệ SQL (MySQL)

- Backend: Sử dụng Spring Boot để viết API, Sử dụng JPA để mapping class (cách tiếp cận hướng đối tượng) trong code với table (cách tiếp cận hướng quan hệ) của SQL.

- Front end: Dựng giao diện sử dụng HTML, CSS. Tương tác với Backend API bằng cách sử dụng library jQuery (cụ thể là AJAX)

3) Loại giao diện của sản phẩm phần mềm của bạn là gì? Để loại nào?

Xây dựng phần mềm dựa trên nền web, dựng giao diện sử dụng HTML, CSS, JS

6) Bạn đã chọn phương tiện nào để triển khai (realization)sản phẩm phần mềm của mình và tại sao?

?????????????

7) Việc phát triển sản phẩm của bạn theo trình tự nào?

// phân tích thiết kế

- Xác định các usecase của hệ thống

- Từ đó xác định các thực thể cần có của hệ thống

- Xác định mối quan hệ giữa các thực thể

- Xác định các thuộc tính của thực thể

- Xác định các hành vi của đối tượng

🡪 Xây dựng nên các class cho hệ thống

- Xây dựng database dựa trên việc xác định thực thể ở trên

// code backend

- Tạo package entity -> tạo các class thực thể ở trên

- Tạo các interface repository để tương tác với csdl

- Tạo controller là nơi chứa các API

- Trong mỗi API, service tương ứng sẽ được gọi để thực thi business logic của API đó

// code frontend

- Tạo giao diện sử dụng HTML CSS

- Với mỗi page, tạo các call API lên backend để tương tác dữ liệu (sử dụng AJAX)

// kiểm thử hệ thống

Thực hiện kiểm thử hộp đen dựa trên các usecase của hệ thông -> phát hiện bug -> fix bug

8) Sản phẩm phần mềm của bạn đã được kiểm tra như thế nào?///nôm na là kiểm thử hộp trắng hay hộp đen ntn

- Với mỗi use case, tiến hành kiểm thử hộp trăng.

Ví dụ.

\* Use case: Làm bài kiểm tra. Check các case sau:

- Lấy bài -> Không chọn đáp án nào -> Thoát khỏi trình duyệt

- Lấy bài -> Tick đáp án -> Thoát khỏi trình duyệt

- Lấy bài -> Tick đáp án -> Nộp bài

- Lấy bài -> Không tích đáp án nào -> Nộp bài

- Lấy bài -> Tick đáp án. Đơi cho time đếm về 0

- Lấy bài -> Không tick đáp án nào -> Đợi cho time về 0

\* Use case: Handle bài kiểm tra

- Check case pass

- Check case fail

- Check kết quả từng câu trong bài kiểm tra

……….

9) Bạn đã sử dụng những phương pháp đánh giá nào và tại sao?

- Sử dụng kiểm thử hộp trắng để đánh giá. Bởi đây là phương pháp kiểm thử thường được sử dụng trong kiểm thử tích hợp và kiểm thử hệ thống. Cách tiếp cận này dựa trên requirement của hệ thống để đánh giá hệ thông. Do đó việc đánh giá chất lượng của phần mềm mang tính tổng quát và dễ tiếp cận với nghiệp vụ hơn. Kết quả của cách tiếp cận kiểm thử này tường minh hơn, người ngoài ngành :v dễ nắm bắt về các case hơn.