Câu 5:

A: Câu chọn phương án phát triển phần mềm:

AGILE

There are several reasons that why **I choose Agile** methodologies used in software development:

* Flexibility: Designed to be flexible and adaptable to changing requirements and priorities. This allows teams to respond quickly to changes in the business environment or user needs.
* Faster time-to-market: Prioritize delivering working software in short iterations, which allows teams to get feedback from stakeholders and adjust quickly. This can help reduce time-to-market and ensure that the product meets user needs.
* Improved collaboration and communication: Emphasize collaboration among team members and with stakeholders. Regular stand-up meetings, sprint reviews, and retrospectives provide opportunities for team members to communicate and work together effectively.
* Continuous improvement: Emphasize continuous improvement, with regular retrospectives and feedback loops that help teams identify areas for improvement and implement changes.
* Increased quality: Prioritize quality through practices such as automated testing, continuous integration, and continuous delivery. This can help ensure that the software is of high quality and meets user needs.
* Customer satisfaction: Focus on delivering software that meets customer needs, through regular feedback and collaboration. This can help ensure that the final product is aligned with customer expectations and leads to higher customer satisfaction.
* **Overall,** Agile methodologies can help development teams work more efficiently, collaborate effectively, and deliver high-quality software that meets user needs.

WATERFALL:

* Well-defined requirements: If the project has well-defined, static requirements that are unlikely to change, the Waterfall model may be appropriate. The linear, sequential approach of Waterfall works well in situations where the requirements are clear and unlikely to change.
* Predictability: The Waterfall model provides a high level of predictability and control over the development process. The linear, sequential approach of Waterfall allows for better control over the project schedule, budget, and scope.
* Simplicity: The Waterfall model is relatively simple and easy to understand. This makes it a good choice for smaller projects with a limited scope.
* Documentation: The Waterfall model places a strong emphasis on documentation, which can be beneficial in situations where detailed documentation is required, such as in highly regulated industries.
* Clear milestones: The Waterfall model has clear, defined milestones, which can make it easier to measure progress and ensure that the project is on track.

B: Testing:

🡪 Kind of testing I would suggest the team do:

- Functional testing: test functional if it appropriate or not.

- Non-functional testing: Is mobile application and website fast, color acceptable, easy to browse or not.

- Unit testing: test class, method in code, is it running right or wrong.

- Integration testing.

- System testing.

- Release testing.

- User acceptance testing.

NOTE:

Nhược điểm WATER FALL :

* Waterfall model has some significant drawbacks, particularly in situations where requirements are likely to change, or where collaboration and flexibility are important. The lack of feedback loops in Waterfall can make it difficult to respond to changing requirements, and the linear, sequential approach can make it difficult to course-correct if problems are discovered later in the development process.

Lack of flexibility: The Waterfall model is a linear, sequential approach that does not allow for much flexibility or adaptation to changing requirements or circumstances. Once a phase is complete, it's difficult to make changes without starting the entire process again.

Limited stakeholder involvement: The Waterfall model does not prioritize stakeholder involvement or collaboration, which can lead to a lack of feedback or misunderstanding of requirements. This can lead to the delivery of a product that does not meet the needs of the stakeholders.

Long feedback loops: The Waterfall model has long feedback loops, with testing and validation being conducted at the end of the development cycle. This can lead to a delay in detecting and addressing issues, which can result in longer development times and higher costs.

High risk: The Waterfall model carries a higher risk of project failure, as issues may not be detected until the end of the development cycle, which can lead to costly rework or the delivery of a product that does not meet requirements.

Difficulty in managing large projects: The Waterfall model is not well-suited to managing large, complex projects, as the linear, sequential approach can make it difficult to manage dependencies and coordinate multiple teams.

Nhược điểm Agile:

Lack of predictability: Agile methodologies are designed to be flexible and adaptable to changing requirements, which can make it difficult to predict timelines and project outcomes. This can be challenging for stakeholders who need to plan and budget for projects.

Lack of documentation: Agile methodologies prioritize working software over documentation, which can make it difficult to maintain a clear record of project requirements, decisions, and outcomes. This can be a problem in highly regulated industries where detailed documentation is required.

Emphasis on collaboration: While collaboration is an important part of Agile methodologies, it can sometimes lead to scope creep and a lack of clarity around project goals and deliverables.

Increased management overhead: Agile methodologies require more management overhead than some other development approaches, as they rely heavily on team collaboration and communication. This can be challenging for organizations that are used to a more hierarchical management structure.

Technical debt: Agile methodologies prioritize delivering working software quickly, which can sometimes lead to technical debt. Technical debt refers to the practice of taking shortcuts or making trade-offs in the development process that can lead to problems down the line.

**PE SU22 HL (Cần học hỏi!!!!)**

Q5:

Despite the software development methodology you have chosen, your manager requires you  
to choose the LEAN methodology to apply to this project.  
a. Should you agree with this requirement or not? If NO, please give the appropriate  
explanation for the WHY question. If YES, give your proposed ideas to require end  
users and another team that could assist you in choosing the LEAN methodology to  
develop this application. (1.5 points)  
b. What kind of testing would you suggest the team do? (1 point)

a. It depends on the specific project requirements and constraints. If the project requires a lot of experimentation and frequent changes, then **LEAN** methodology might be more appropriate. However, if the project has a clear set of requirements and a fixed scope, then Agile methodology might be a better fit. If the manager insists on using LEAN methodology, I would propose involving end users and other team members in the decision-making process to ensure that the methodology aligns with the project goals and constraints.

(**Waterfall**: It depends on the specific requirements and constraints of the project. If the project does not require a lot of testing and frequent changes, the Waterfall approach may be more suitable. However, if the project has a clear set of requirements and a fixed scope, then Agile methodology might be a better fit. Then **Finally** I choose **Agile methodology** because: (Kể ra ở trên)).

(Nó phụ thuộc vào các yêu cầu và ràng buộc cụ thể của dự án. Nếu dự án không yêu cầu kiểm thử nhiều và thay đổi thường xuyên thì cách tiếp cận Thác nước có thể phù hợp hơn. Tuy nhiên, nếu dự án có một bộ yêu cầu rõ ràng và phạm vi cố định, thì phương pháp Agile có thể phù hợp hơn. Sau đó, cuối cùng tôi chọn phương pháp Agile vì:)

b. The team should focus on automated testing to ensure that the app is reliable, scalable, and secure. This should include unit testing, integration testing, and end-to-end testing. In addition, the team should conduct user acceptance testing to ensure that the app meets the requirements and expectations of the end users.

**PE RETAKE SP22 9h30**

Q5:

**(Agile methodology)**

1. Yes, I would agree with the manager's requirement to apply Agile Software Development to this project. Agile methodology is a good fit for this project as it involves frequent communication with stakeholders and the ability to quickly adapt to changing requirements, which is essential in a project that requires continuous interaction with customers.

(Có, tôi đồng ý với yêu cầu của người quản lý là áp dụng Phát triển phần mềm linh hoạt cho dự án này. Phương pháp Agile rất phù hợp với dự án này vì nó liên quan đến việc giao tiếp thường xuyên với các bên liên quan và khả năng thích ứng nhanh với các yêu cầu thay đổi, điều cần thiết trong một dự án đòi hỏi sự tương tác liên tục với khách hàng.)

To require end-users and the team to use Agile, I would propose the following ideas:

* Conduct a training session on Agile principles, values, and practices to ensure everyone on the team has a shared understanding of Agile.
* Involve end-users in the development process by conducting regular user feedback sessions to gather feedback and ensure that the team is building features that meet their needs.
* Hold daily stand-up meetings to keep everyone on the same page, identify roadblocks, and ensure that the team is working towards the same goals.

🡪Use Agile tools such as user stories, sprint planning, and retrospectives to ensure that the team stays focused on delivering high-quality features that meet end-user needs.

b. **(nên lấy câu trả lời test này !! )**

**Since testing is an essential part of Agile software development, I would suggest the team implement the following types of testing:**

**-Unit testing: to ensure that each individual piece of code works as expected.**

**-Integration testing: to test how different components of the system work together.**

**-Acceptance testing: to ensure that the system meets the end-user's requirements and can be used in the intended way.**

**-Regression testing: to ensure that changes to the system have not caused any unintended side-effects or bugs in existing functionality.**

**🡪 In addition to these types of testing, I would also suggest using test automation tools to ensure that testing can be performed quickly and reliably during each iteration. This will help the team to deliver high-quality features faster and with fewer defects.**

(Vì thử nghiệm là một phần thiết yếu của quá trình phát triển phần mềm Agile, tôi khuyên nhóm nên thực hiện các loại thử nghiệm sau:

-Unit testing: để đảm bảo rằng từng đoạn mã riêng lẻ hoạt động như mong đợi.

-Kiểm thử tích hợp: để kiểm tra xem các thành phần khác nhau của hệ thống hoạt động cùng nhau như thế nào.

-Kiểm tra chấp nhận: để đảm bảo rằng hệ thống đáp ứng các yêu cầu của người dùng cuối và có thể được sử dụng theo cách đã định.

-Regression testing: để đảm bảo rằng những thay đổi đối với hệ thống không gây ra bất kỳ tác dụng phụ hoặc lỗi ngoài ý muốn nào trong chức năng hiện có.

 Ngoài các loại thử nghiệm này, tôi cũng khuyên bạn nên sử dụng các công cụ tự động hóa thử nghiệm để đảm bảo rằng thử nghiệm có thể được thực hiện nhanh chóng và đáng tin cậy trong mỗi lần lặp lại. Điều này sẽ giúp nhóm cung cấp các tính năng chất lượng cao nhanh hơn và ít lỗi hơn.)

FINAL RETAKE SP22 7h30

**Waterfall**

A: I would not agree with the manager's request to use the Waterfall software development methodology. Waterfall is a more rigid and sequential approach that may not be well-suited for this project, given the need for flexibility and collaboration with stakeholders. To convince the manager to use Agile, I would emphasize the benefits of continuous feedback and collaboration, as well as the ability to adapt to changing requirements. I would also suggest involving end users and other team members in the decision-making process, to ensure that everyone is on board with the chosen methodology.

(Tôi sẽ không đồng ý với yêu cầu của người quản lý về việc sử dụng phương pháp phát triển phần mềm Waterfall. Thác nước là một cách tiếp cận tuần tự và cứng nhắc hơn, có thể không phù hợp lắm với dự án này, do nhu cầu về sự linh hoạt và hợp tác với các bên liên quan. Để thuyết phục người quản lý sử dụng Agile, tôi sẽ nhấn mạnh lợi ích của việc cộng tác và phản hồi liên tục, cũng như khả năng thích ứng với các yêu cầu thay đổi. Tôi cũng khuyên bạn nên để người dùng cuối và các thành viên khác trong nhóm tham gia vào quá trình ra quyết định, để đảm bảo rằng mọi người đều đồng ý với phương pháp đã chọn.)

B: Given the importance of security and data handling in this system, I would suggest conducting thorough testing of the system's security measures, as well as its ability to handle large amounts of data and perform complex data analysis. This could include penetration testing, load testing, and stress testing, among other types of testing. Additionally, functional testing would be necessary to ensure that the system meets the requirements of the manager and other stakeholders.