**Solution(9.5->10đ)**

Question 1:

**Requirements characteristics:**

* Reliability:
* The system must provide high-quality service and avoid any potential technical or user problems after implementation.
* The requirements are clear but not detail and this project is possible to run.
* Types of requirements:
* The requirements contains both non-functional requirements and functional requirements, the requirements are clear and not ambiguous:
* Functional Requirements: These describe what a system must do, specifying its functions and capabilities.
* Non-Functional Requirements: These describe how well the system must perform its functions, specifying its performance, reliability, usability, and other characteristics.
* How often the requirements can change:
* The requirements are not fully defined yet and will be refined in the coming weeks or months.
* Can the requirements be defined at an early stage:
* Yes, the high-level vision for the web application has been defined by the business stakeholders.

**Development team:**

* Team size:
* The team consists of 6 developers and 2 QA.
* Level of understanding of user requirements by the developers**:**
* The developers have not received detailed requirements yet, but the business stakeholders will help refine or define the user needs.

**User involvement in the project:**

* Users are participants in the product experience, they can make suggestions for the product to improve according to user needs, so we can conclude that user participation in this project is large.

**Software development methodology:**

* 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.
* Based on the characteristics of the project, I would suggest using the **Agile** software development methodology.

Question 2:

* Functional Requirements:
* Allow managers and employees to set performance goals and track progress.
* Allow managers to provide feedback on employee performance and employee to give feedback on their performance.
* Allow managers to conduct formal performance evaluations and provide ratings or scores, or project completion rates.
* Allow managers to access employee competencies, such as technical skills, communication skills, or leadership abilities.
* Non-Functional Requirements:
* Performance: The system should provide a high-quality service and avoid any potential technical or user problems after implementation.
* Security: The system must protect and faculty information and prevent unauthorized access.

Question 3:

* User Story:
* As a manager, I want to be able to provide feedback on employee performance and employee to give feedback on their performance, so that I can improve the system on according to employee needs.
* As a manager, I want to be able to access employee competencies, such as technical skills, communication skills, or leadership abilities, so that I can provide accurate and up-to-date competencies to employee.

Question 4:

Home Screen

Reviews and Ratings

Accounts Management

Salary Management

Feedback

Services

--------------------------------------------------- **Release 1** -----------------------------------------

Enter Confirm information.

List font services.

sign up or log in using mobile number, email, or Facebook.

their salary details, request and check their income tax details.

provide feedback on their experience.

Allow review and ratings.

--------------------------------------------------- **Release 2** -----------------------------------------

Allow Modify all information.

List of all the services offered.

Provides categories for services.

leave a review and rating for each other or payments.

view their income tax information and report any discrepancies.

rate their services and leave comments or suggestions and view review.

---------------------------------------------------- **Release 3** -----------------------------------------

Loyalty program: Customers could earn points or rewards for their bookings and receive discounts or special offers based on their loyalty.

Modify confirm approval.

Social media integration: Customers could share their bookings, reviews, and feedback on social media platforms such as Facebook, Twitter, and Instagram

Shows promotions and the newest offers and discounts for users.

Question 5:

* Assumptions:

1. The competency assessment feature will accurately measure the employee's skills and abilities.
2. The competency assessment feature will provide consistent results across different assessors.
3. The competency assessment feature will be easy to use and understand for both managers and employees.

* Classification:

1. High impact if wrong, High Probability of it being wrong.
2. High impact if wrong, Low Probability of it being wrong.
3. Low impact if wrong, High Probability of it being wrong.

* Explanation:

1. If the competency assessment feature does not accurately measure the employee's skills and abilities, it could lead to incorrect evaluation of their performance and development needs. This could result in wasted resources on training or promotion of employees who are not yet ready or demotivate and discourage employees who may have potential. The high probability of it being wrong is due to the complexity and subjectivity of the evaluation process, which can vary depending on the assessor's perspective and expertise.
2. If the competency assessment feature does not provide consistent results across different assessors, it could result in unfair evaluations and inconsistent treatment of employees. This could lead to mistrust and dissatisfaction among employees and undermine the credibility of the performance monitoring system. The low probability of it being wrong is due to the possibility of using standardized assessment criteria and training assessors on how to apply them consistently.
3. If the competency assessment feature is not easy to use and understand for both managers and employees, it could lead to incorrect or incomplete assessments, or low adoption rates of the feature. This could result in poor quality data, inaccurate evaluations, and limited use of the feature, undermining its potential benefits. The high probability of it being wrong is due to the possibility of having different levels of technical skills and familiarity with the feature among users.

Question 6:

* 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.
* 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.
* 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.