**Question 1:**

1. Requirements characteristics
   * Reliability:
     + It was stated clearly above that there is a problem that needs to be overcomed, and an application is required to overcome this problem.
     + The project requirements are well-defined and possible.
     + It can immedate run while the project finished.

=> The project is expected to be highly reliable.

* + Types and number of requirement:
    - The software requirements include both *functional requirements* and *non-functional requirements*. All of them defined clearly and not confusing.
    - There are more than 8 requirements that was listed above. It include 6 functional requirements and 2 non-functional requirements.

=> Types and number of requirements defined this project is not too complex for our team.

* + Frequency of requirement may change:
    - The requirements that was mentioned above, is just some features of this appication.
    - In the process of project, some of features can be modified and changed to meet with the requirements of customer.

=> The requirements may change regular.

* + Determination of requirements at an early stage
    - Some of requirements was defined above but it isn’t enough to build a completed system.
    - The organization can be added or removed some of features in the process of project.

=> It is well-defined but not enough.

1. Development team
   * Team size:
     + The situation above metioned our team have 9 people.
     + It is *6 developers, 2 QA* and a project owner who is me.

=> It is a average team size and enough to build a project that was not too complex.

* + Level of understanding of user requirements by the developers:
    - All of requirements defined clearly above and our member can understand.
    - The organization can provide additional resources and information when needed.

=> Our team can easy to understand and build a appication that meets the requirements.

1. User involvement
   * The situation mentioned “*The organization had contracted with a local company to provide additional resources when needed*.”
   * We have a contact with organization to communicate and give the feedback about the project.

=> The user involvement is highly.

Based on the characteristics mentioned in the context of the software development project, it can be concluded that the **Agile/Scrum methodology** is the best approach to use. The Agile/Scrum model is iterative and incremental, prioritizing flexibility, collaboration, and customer feedback at every stage of development. It also allows for customers to release the product early and gather feedback from users, which gives the development team an opportunity to apply user feedback into future iterations of the product. This customer-centric approach ensures that the final product meets the requirements of its users. Overall, the Agile/Scrum methodology is well-suited for this software development project and will likely result in a high-quality end product.

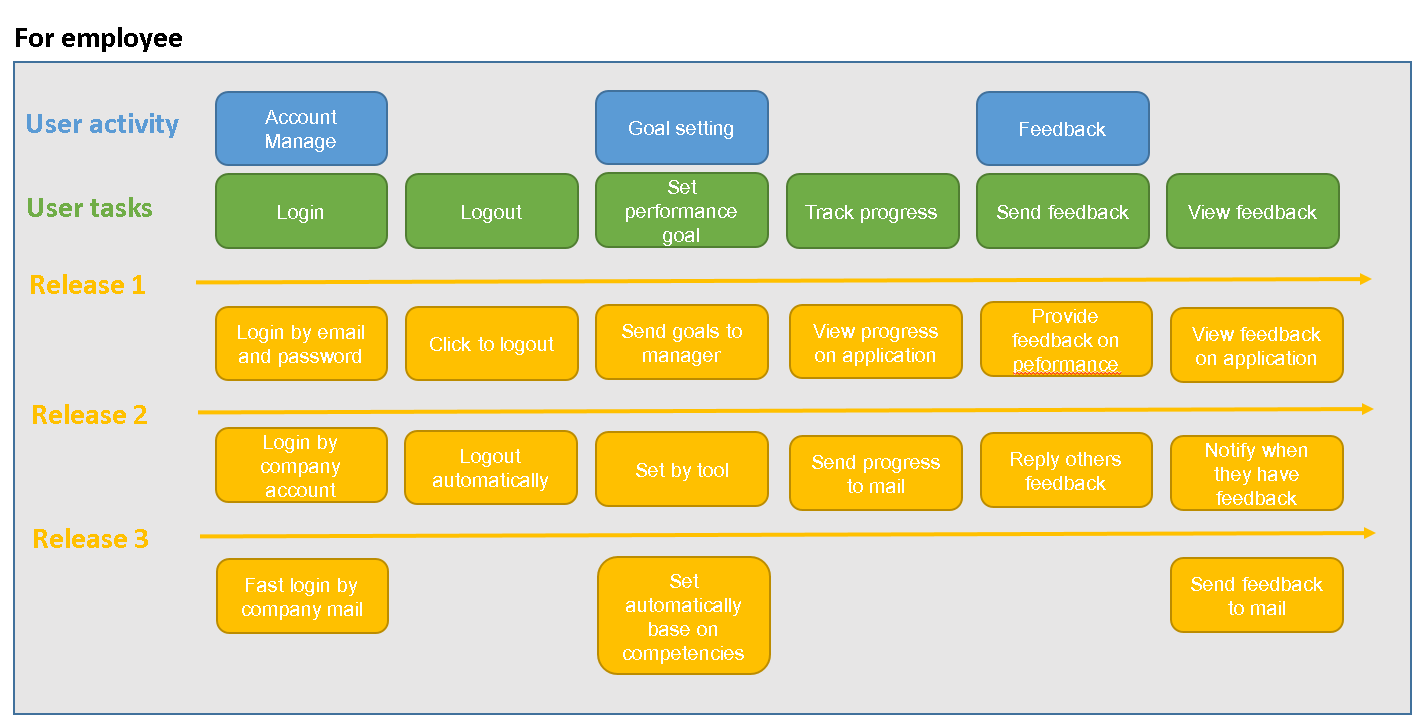
**Question 2:**

* The four functional requirements of system are:
  + Managers and employees can set performance goals and track progress.
  + Managers can provide feedback on employee performance and employees can give feedback on their performance.
  + Managers can conduct formal performance evaluations and provide ratings or scores based on the employee's performance.
  + Managers to assess employee competencies, such as technical skills, communication skills, or leadership abilities
* The two non-functional requirements of system are:
  + The system should maintain employee privacy by ensuring that only authorized individuals can access employee performance data.
  + The module should be able to handle a growing number of users and tasks without degrading performance or reliability.

**Question 3:**

* The two user stories for this system are:
  + As a manager, I want to set goals for employees so that I can set peformance goals that agree with each employee and track progress.
  + As a employee, I want to provide feedback on my performance so that I can send feedback for my manager and manager can track my progress.

**Question 4:**



**Question 5:**

The three assumptions regarding the competency assessment feature are:

* **Goal setting** feature is high impact if wrong, low probability of it being wrong.
  + It is needed for the process of any project.
  + It help employees can track their goals.
* **Feedback** feature is low impact if wrong, high probability of it being wrong.
  + Beside the feedback in system, they also have daily meeting. Face to face feedback is more efficiently than message feedback.
* **Development plans** feature is high impact if wrong, low probability of it being wrong.
  + Managers and employees must create development plans for any tasks.
  + They need follow this plan to finish any tasks better.
  + If it is wrong, employee can miss their progress easily.

**Question 6:**

I recommend that the team use **black-box testing** as it allows for user involvement in identifying and providing feedback, specifically with regards to the system's usability. Additionally, there is *no mention of the tester's knowledge or experience* in the project description, making black-box testing a suitable option. Furthermore, this type of testing does not require specialized expertise from the analyst, as detailed technical knowledge of the system is not necessary.