**Question 1: Software development methodology I would suggest for this situation is Scrum/Agile**

1. Requirements characteristics
   * Reliability:
     + The requirements are defined quite clearly but they are not enough and still have some missing points that need to discover later in the development process.
     + The project requirements are well-defined and possible, it is efficient and usable.
     + It can immediate 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 but they are not *detail*

=> Types and number of requirements defined this project is quite complex for our team.

* + Frequency of requirement may change: There maybe many changes during the development process according to the information.

=> 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.

CONCLUSION: with a project that is highly reliable, complex with multiple requirements, well-defined but not enough and not detail, and the requirements may change regular, SCRUM/AGILE is eligible.

1. Development team
   * Team size: 4-6

=> SCRUM/AGILE works best in a cross-functional team of 4 to 9 developers working on a medium to large-scale project

* + Level of understanding of user requirements by the developers: The project is undertaken by IT department of FU and other departments commit to sending employees to join the project team to support => high level of understanding

CONCLUSION: the size of team and the skill, experience and understanding of the team is suitable for AGILE/SCRUM

1. Customer involvement
   * The situation mentioned “*The project is undertaken by IT department of FU and other departments commit to sending employees to join the project team to support*.”

=> The customer involvement is highly and likely to feedback the phases of the project => AGILE/SCRUM

* + Time constraints: “put into use within 3 months and the project needs to be completely completed within 9 months” => quite long time and many phases => AGILE has many phases to release products

**CONCLUSION**: 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: Levels of tests I suggest the team use are:**

- Unit testing (performed by the developers): If applying AGILE/SCRUM, each developer will take some tasks, they need to use Unit test to ensure other tasks are ok

- Integration Testing (performed by testers): I suppose the team will have testers

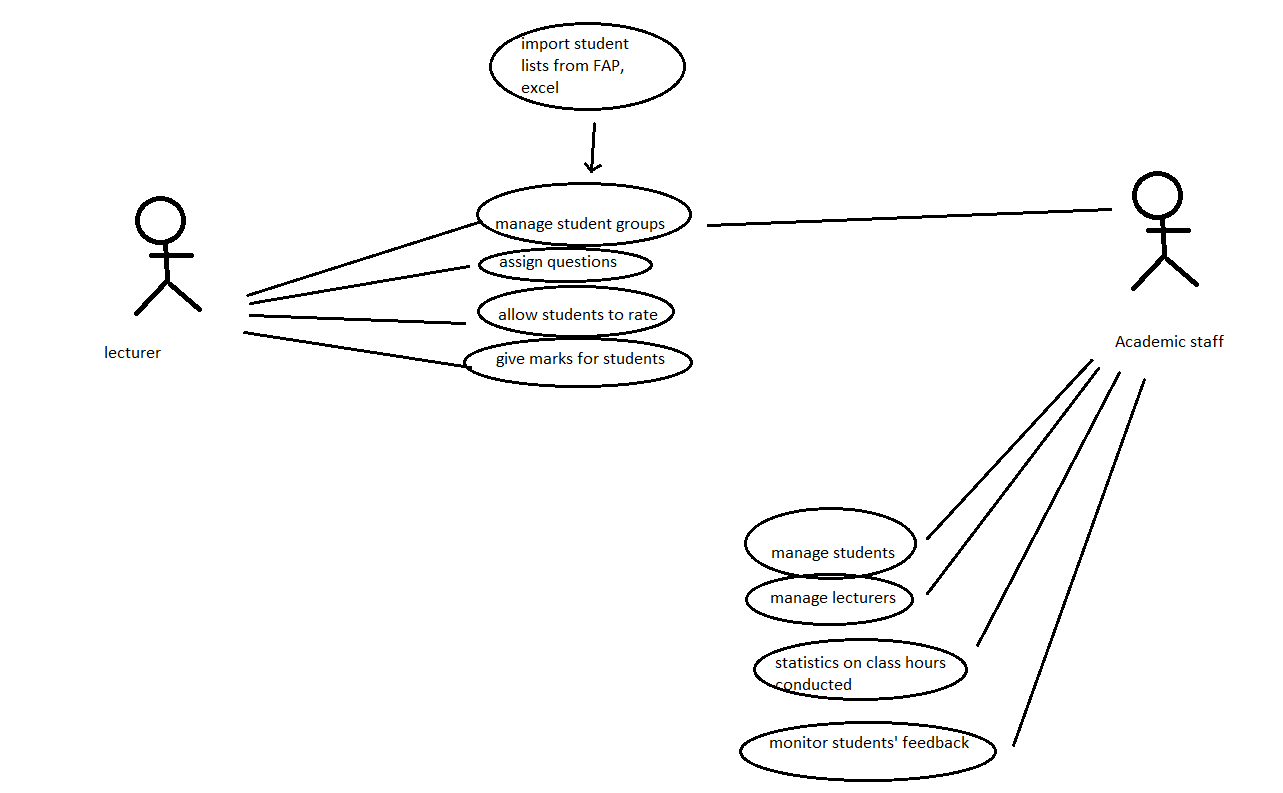
- Acceptance Testing (is basically done by the user or customer and other stakeholders): in the end of each phase, the product need to gather feedbacks, so the acceptance testing is necessary.

About test types, I suggest the team use functional testing, non-functional testing and also white box testing

**Question 3:**

1. 4 functional requirements:
   * Lecturers can import student lists
   * Lecturers can create reports on student activities
   * Students can view constructive questions
   * Users are allowed to log in with the FU’s email account on the Gmail platform.
2. 2 non-functional requirements:
   * The system needs to ensure information security.
   * The system needs to ensure high performance and reliability, requiring little training time to use.

**Question 4:**



**Question 5: 4 functional test case**

**-** Test if lecturers import student list successfully or not?

- Test if lecturers assign questions successfully or not?

- Test if someone does not login by FPT mail successfully or not?

- Test if Lecturers can create reports on student activities successfully or not?

Mostly use black-box to test.

**Question 6:**

- As a lecture, I want to import student lists so that I can manage the class and create homework for students

- As a student, I want to view constructive questions so that I can answer them and get marks from lecturers

**Question 7:**

