Q1:   
For the development of the FU Next Examination system (FU-NextExam) given the rapid timeline and expectation for iterative and incremental delivery, the Agile software development methodology, particularly the Scrum framework, is recommended.

Project Characteristics:

* User, Customer, and Team Dynamics:
  + The system will serve a diverse group of users, including lecturers, students, and lecturer leaders, each with unique requirements that must be clearly understood and quickly addressed.
  + The development team is comprised of 4-6 experienced IT personnel alongside contributors from other departments, implying a need for a method that supports team dynamics and leverages various skill sets.
* Requirements Characteristics:
  + FU-NextExam requirements may not be fully defined upfront and are subject to change, thus a methodology that embraces requirement evolution is crucial.
  + The system requires high quality and security standards, which Agile can assure through continuous testing and integration.
* Time Constraints and Management Expectation:
  + A tight deadline necessitates a development model that allows for concurrent phases of planning, development, testing, and revisions.
  + Management expects quick and tangible results, which Agile’s sprint cycles can deliver, providing frequent progress updates and product increments.

Development Model Choice:

Scrum fits the identified factors well as it:

* Encourages regular reflection on how to become more effective, allowing the team to adjust behaviors accordingly.
* Utilizes time-boxed sprints to divide the work into manageable chunks, which aligns with the three-month release target.
* Emphasizes daily communication and collaboration through rituals like Daily Stand-Ups and Sprint Reviews, ensuring the team stays aligned and bottlenecks are addressed promptly.

Considering these points, Scrum’s iterative development cycles, emphasis on user feedback, and ability to accommodate changing requirements will likely lead to the successful delivery of SBMS within the specified timeframe and to the satisfaction of all stakeholders.

Q2.

Suggested Testing Types and Levels for FU-NextExam

Unit Testing

* Executors: Developers
* Timing: Post-development of individual functions
* Objective: Verify the correctness of each unit of code in isolation

Integration Testing

* Executors: Development Team or Integration Testers
* Timing: Post-completion of module development
* Objective: Ensure that integrated modules operate cohesively

System Testing

* Executors: Quality Assurance Team
* Timing: After integration and before UAT
* Objective: Confirm that the system as a whole meets specified requirements

User Acceptance Testing (UAT)

* Executors: End Users/Stakeholders
* Timing: Before final deployment
* Objective: Validate the system against user requirements and expectations

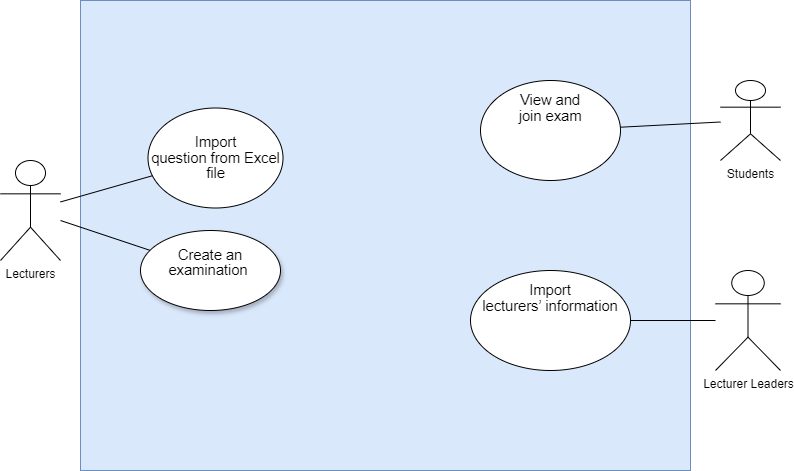
Q3.

Functional requirements:

* Import question from Excel file: the system shall allow the lecturers upload the question form their Excel file on their computer or laptop
* Create an examination: the system shall allow the lecturers to create the examination
* View and join exam: the system shall allow the students to view a list of exam and join the exam that the lecturers has provide on the system
* Import lecturers’ information: the system shall allow the lecturer Leaders to import information of the lecturers

Non-funtional requirement:

* Performance: The system shall process route registration and modification requests within 2 seconds under normal operation conditions.
* Security: The system shall enforce role-based access controls to ensure users can only access features relevant to their permissions.

Q4: 

Q5: Functional Test Cases for FU Next Examination system (FU-NextExam)

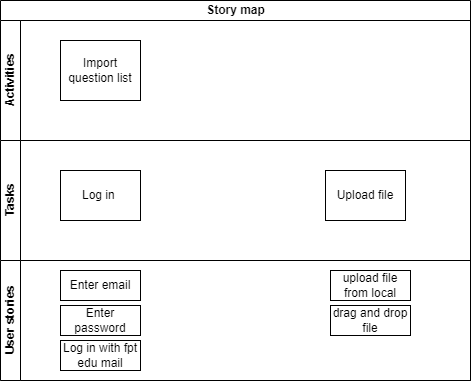
* Test Case 1: Import question from Excel file
  + Test Case ID: TC\_FUNE\_IQE\_01
  + Test Description: Verify that a lecturer can successfully import a question from an Excel file on their computer or laptop
  + Pre-conditions: Lecturer is logged in with the appropriate privileges.
  + Test Steps:
    - Navigate to the ‘Question’ page.
    - Click the “upload” to upload the appropriate Excel file
    - Click the 'Submit' button.
  + Test Data: an arbitrary Excel file, and an appropriate Excel file
  + Expected Result: The system confirms the questions has been uploaded and display it on the question page
  + Actual Result: [To be filled during testing]
  + Status: [To be filled during testing]
  + Remarks: N/A
* Test Case 2: Create an examination:
  + Test Case ID: TC\_FUNE\_CAE\_01
  + Test Description: Verify that a lecturer can successfully create an examination
  + Pre-conditions: Lecturer is logged in with the appropriate privileges.
  + Test Steps:
    - Navigate to the ‘Examination’ page.
    - Click the “Create” button
    - Click the 'Submit' button.
  + Test Data: N/A
  + Expected Result: The system confirms the examination has been created and display it on the exam page
  + Actual Result: [To be filled during testing]
  + Status: [To be filled during testing]
  + Remarks: N/A
* Test Case 3: View and join exam:
  + Test Case ID: TC\_FUNE\_VAJ\_01
  + Test Description: Verify that a student can successfully view his/her exam list and join the exam
  + Pre-conditions: Student is logged in with the appropriate privileges.
  + Test Steps:
    - Navigate to the ‘Examination’ page.
    - Click the “View” button to see the exam list
    - Click the “join” button to join a current exam
  + Test Data: N/A
  + Expected Result: The system confirms the exam list has been display on the screen and student can join them
  + Actual Result: [To be filled during testing]
  + Status: [To be filled during testing]
  + Remarks: N/A
* Test Case 4: Import lecturers’ information:
  + Test Case ID: TC\_FUNE\_ILI\_01
  + Test Description: Verify that a Lecturer Leader can successfully import information of an arbitrary lecturer
  + Pre-conditions: Lecturer Leader is logged in with the appropriate privileges.
  + Test Steps:
    - Navigate to the ‘Lecturer informations’page.
    - Click the “Import” button
    - Click the “Submit” button
  + Test Data: a lecturer name’s “Nguyen Van A”, age: 46, main major: Artificial Intelligent
  + Expected Result: The system confirms the lecturer information has been imported and display it on the screen
  + Actual Result: [To be filled during testing]
  + Status: [To be filled during testing]

Q6.

* User Story 1: View and join exam
  + As a student, I want it is feasible to view the up-coming test so that I can finish that exam before deadline with the highest quality
  + Acceptance Criteria:
    - The user must be logged in with student privileges.
    - The system should display a list of existing examination to select for participant.
    - The system must allow editing of Departure Time and Route Stops.
    - After join request are submitted, the system should navigate to the exam
* User Story 2: Import lecturers’ information
  + As a lecture Leader, I want to easily mange the lecturer so that my university can be better at teaching
  + Acceptance Criteria:
    - The user must be logged in with lecturer Leader privileges.
    - The system should display a list of existing Lecture and a import button
    - Upon submission, the system should validate the input and give immediate feedback.
    - Once submitted, the new lecturer should be visible in the list of available lecturer information

Q7.

* **Import Question List:**



* **Create a New Exam:**

