**Question 1:**

* **7 functional requirements:**
  1. The system will allow students to register for 4 courses each semester and view report cards from personal computers attached to the campus LAN
  2. Professors will be able to access the system to sign up to teach courses as well as record grades.
  3. Show the list of course offerings for the semester with the information about each course at the beginning of each semester for students
  4. There is period of time that students can change their schedule for each semester.
  5. At the end of the semester, the student will be able to access the system to view an electronic report card.
  6. Professors must be able to access the on-line system to indicate which courses they will be teaching.
  7. The professors will be able to record the grades for the students in each class.
* **3 non-functional requirements:**
  1. The new system will access course information from the legacy database but will not update it. Using an open SQL interface that allows access to the old database (which is an Ingres relational database running on a DEC VAX) from College’s Unix servers.
  2. ~~Course offerings will have a maximum of ten students and a minimum of three students. A course offering with fewer than three students will be canceled.~~
  3. The legacy system performance is rather poor, so the new system must ensure that access to the data on the legacy system
  4. The system must employ extra security measures to prevent unauthorized access.

**Question 2:**

* **Student:**
  1. As a student, I want to view the list of course with the information such as professor, department, and prerequisites to decide register for courses for the coming semester.
  2. As a student, I want to personally view my electronic report at the end of the semester.
* **Professor:**
  1. As a professor, I want to indicate which courses will be teaching, and see which students register for these courses.
  2. As a professor, I want to record and edit the grades of the students in each class I’ve taught.

**Question 3:**

Analyze the requirements:

* Well-known problem: The functions of the new system has been defined completely.
* Well-known solution: We also defined the solution for the new system which is a client-server system and using an open SQL interface to access to the old database.
* Number of requirements has been defined and rarely changes.
* The team consists of 7 people and will be supported by the IT department of the College who understanding the business 🡺 medium team size and the team can be completely understand the requirements.
* The system involves a large number of students and professors
* **The system is enhanced from old system 🡺 Something must be changed**

🡺 In my opinion, I suggest to use ~~Waterfall model~~ for this project. Because everything about the requirement is fairly known ~~and rarely to change~~. This model is also very simple to implement. However, because of enhancing from old system, using new technology will lead to some changes and comparations 🡺 **Incremental model**

**Question 4: Story map for the course registration of the system**

Authentication & Authorization

Register a course

Change the course has been registed

Sign up

Log in

Log out

View course list

Register for a course

Add new courses

Drop courses

Using email to reset the password in case Forgot the password

View course infomation

Course will have max of 10 min of 3 students.

A course fewer than 3 students will be canceled.

Student will be notified when the course be canceled.

Student will be able to choose 4 course for the coming semester

Student will indicate two alternative choices in case cannot be assigned to a primary selection.

Students must be able to access the system during the time to edit

If a course added during the registration process, the student must be notified of the change before submitting the schedule for processing.

Show the bill to the student once the registration process is completed