(Q1) Compare the strengths and weaknesses of Waterfall model and Agile process. (10 points)

The agile process has many pros but first and foremost is equipped to be extremely responsive to change and deal with projects where business requirements can change on the fly or new features will become desired while the project is already under development. The agile process accomplishes this versatility by having teams commit to daily scrum meetings where they present what they've worked on to ensure it meets specification, communicate with business about new features that need to be added, and assign tasks to be worked on for the next sprint. The drawback of agile is that unlike waterfall, these meetings can be time consuming and the uncertainty of what new features may be added can make the development process more ad hoc and on the fly. Waterfall on the other hand aims to linearly perform the task of application development step by step. The pros of waterfall are that everyone is on the same page and working on one thing at a time, the end goal is very clear right from the start of the project and you always know where you are in relation to that end goal. The drawback of this is that waterfall has an incredibly hard time responding to change due to needing to start back at the first stage of development for every new feature added or changed.

(Q2) Object-oriented design/programming has good features to promote reusability, productivity, and maintainability. Describe how these features can be applied to improve reusability, productivity, and maintainability, and what are the potential drawbacks. (10 points)

Object oriented design is an incredible tool to improve reusability, productivity, and maintainability, but also comes with some substantial drawbacks depending on the situation. Firstly compartmentalizing objectives and concepts into objects allows greater reusability because once a concept has been converted into an programmatic object, that object can be reused anywhere in the project that requires code that makes use of that concept and generalized objects can even be used on other projects. Secondly, object oriented design boosts productivity because humans think in terms of objects and concepts rather than in terms bits and bytes. It's much easier and faster for us to code and reason about higher abstractions then it is for us to twiddle bits and bytes at the hardware level, not to mention the reusability and maintainability provided by object oriented design is also a major contributor to productivity. Thirdly, object-oriented design contributes to maintainability due to the principles of OOP: encapsulation, abstraction, inheritance and polymorphism. The abstraction and encapsulation principals in particular help us here as we can update members and internal workings of an object without worrying about breaking things outside of the object, meaning OO design is modular and much easier to maintain. Some potential drawbacks of object oriented design are performance costs due to abstractions as well as the complexity added by working with concepts and needing to learn about what objects you have at your disposal before working on a project.

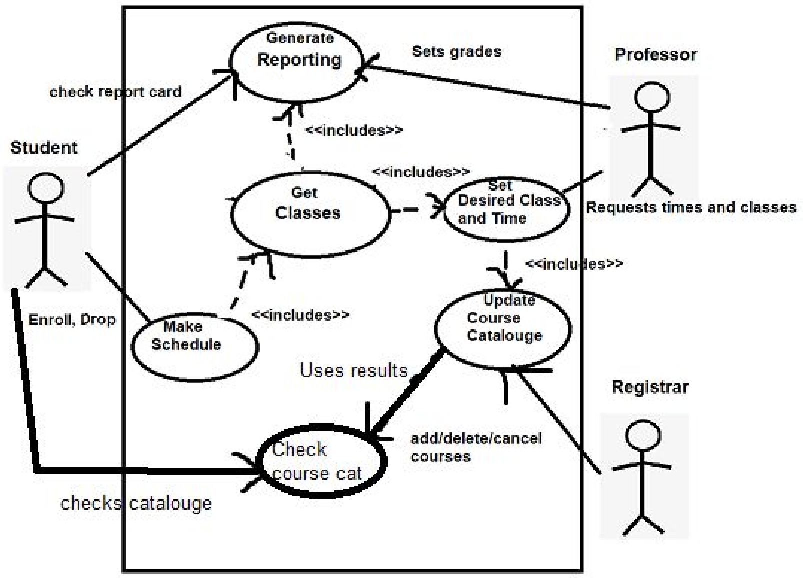
(Q3.) Write a testable “Drop a course” functional requirement for the course registration system. (10 points)

Drop a course	If the user attempts to access the site during the time period in which they are allowed to change their schedule, the user will be able to login to the application and see a list of what course they have currently registered for. There will be a drop button on each course listed that when clicked will create a popup asking the user if they are sure that they want to drop the course. If the user selects “No”, the course will not be dropped and they will be returned to the listing page. If the user selects “Yes” the course will be dropped and disappear from the listing, and a request will be made to the server to drop the user from the course.

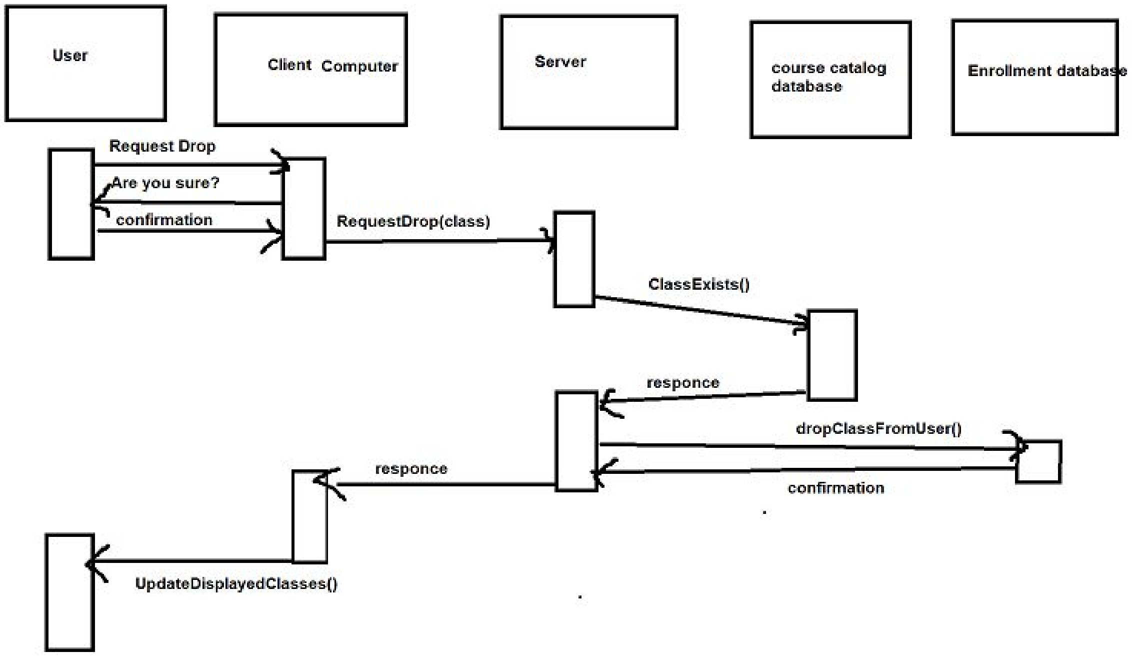
(Q4) Write a testable “usability” non-functional requirement for the course registration system. (10 points.)

Usability	The application must be able to be learned very quickly due to it being used by professors for their classes; must be learnable in an hour. Must be available and running with no errors during the time period during which students are to register, must allow students to access their grade on time. users must be notified when reports are released. Must provide an easy to use interface for registrars to interact with so they can add course offerings.

(Q5) Create a use case diagram for the course registration system (10 points)



(Q6) Create a sequence diagram for the use case “Drop a Course” for the course registration (10 points)


User
Client Computer
Server
course catalog database
Enrollment database
Request Drop
Are you sure?
confirmation
RequestDrop(class)
ClassExists()
responce
dropClassFromUser()
responce
confirmation
UpdateDisplayedClasses()