



Software Use Cases

↳ Defining how
SW is used.

Objectives

- Define Actor -
- Define a use case -
- Interpret the meaning of a use case diagram.
- Explain the relationship between Use Case Diagrams and Use Case Scenarios
- List the items present in a use case scenario
- Construct a use case scenario for a given problem
- Explain how the level of detail in use cases may change throughout the software development process. — Changes

- Turn to your neighbor and answer the following:
 - List attributes of good requirements

Lecture Review

Take a look at the following attributes of good requirements:

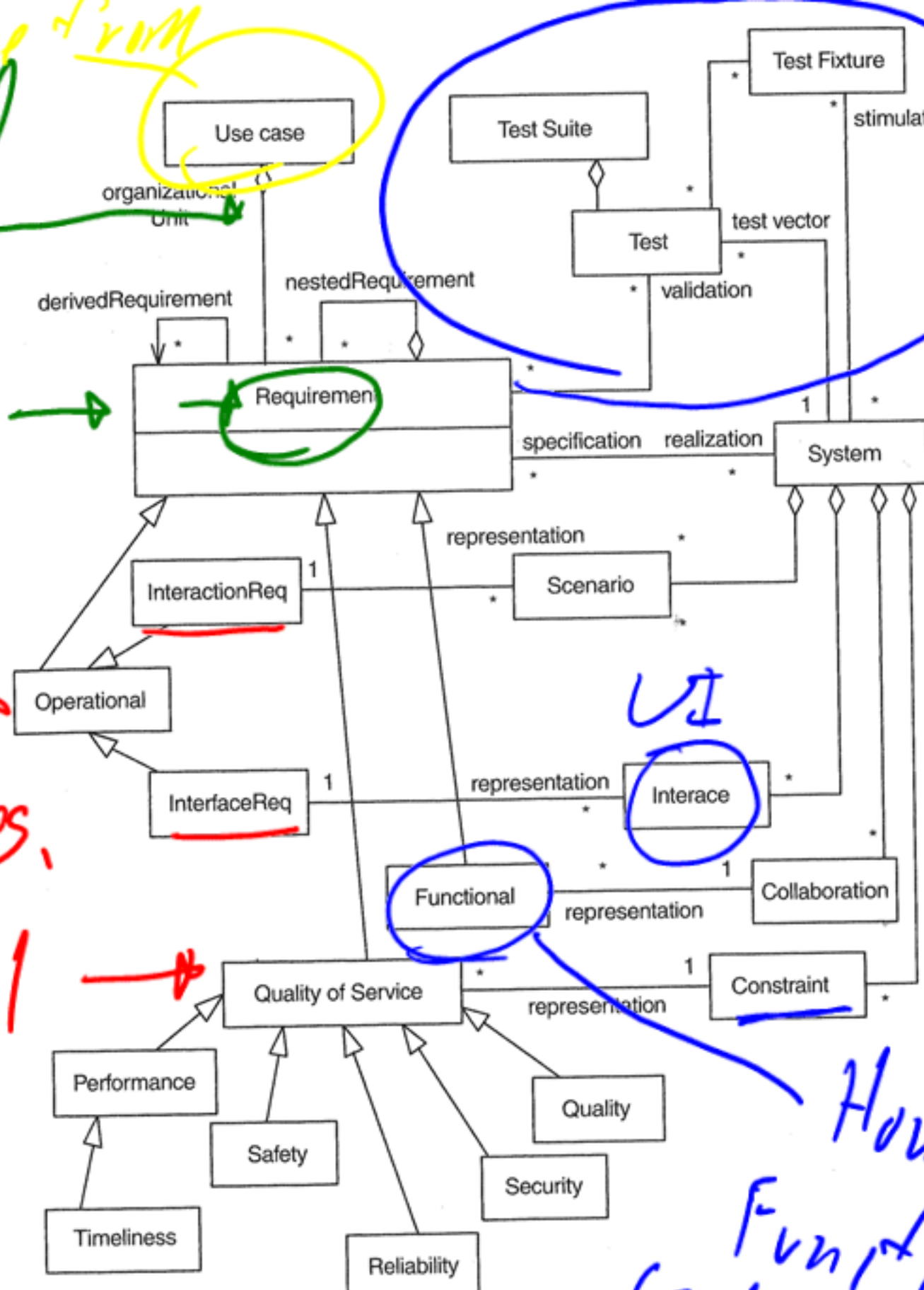
- Specific
- Concise
- Obtainable - Testable
- Unambiguous
- Complete

Categorizing Requirements

All come from
Derived from Use Cases

How SW operates.

How SW works.



Drive testing

How SW Functions (Interaction)



Use Cases and Use case

Diagrams

- Use Case — *Set of all use cases defines the capabilities of the system.*
 - A named capability of a structural entity within a system
 - Explanation of how someone would use the system or a capability of the system
- Use Case Diagram *> Graphical Representation*
 - A diagram showing use cases and the relationship with Actors
 - Actor — *User / Existing System.*
 - An external entity which interacts with the system

Definition

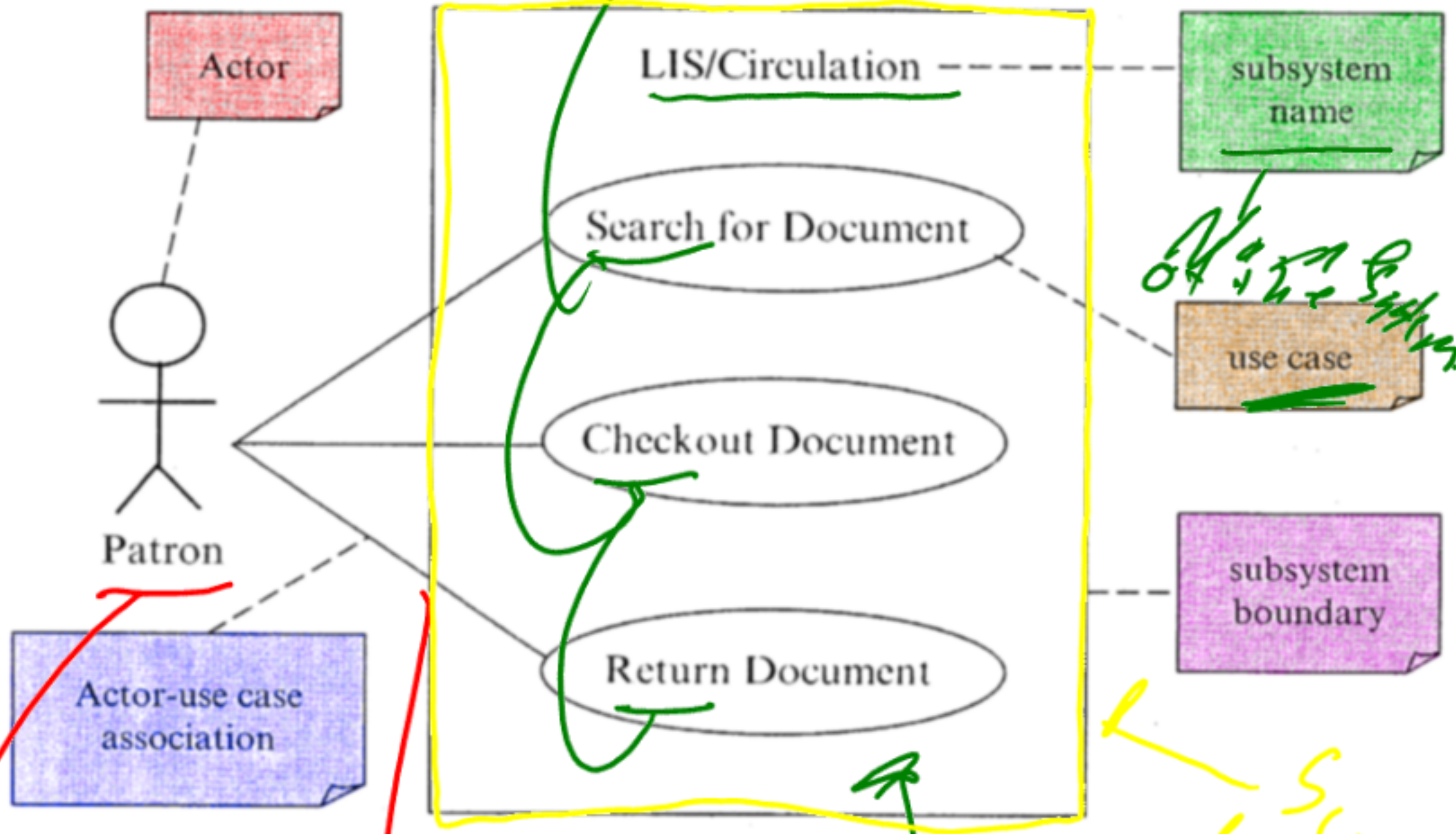
- Actor (Page 282-284 of textbook)
 - An object outside of the scope of the system which has significant interactions with the system
 - Typically users are a good example, but they are not the only example
 - Named with singular nouns

↳ Things

Use Case Diagram

"Library System"

Verbs / Actions



Noun

(verbs)

Use Cases
Scope



In Class Exercise

- Work with your neighbor next to you
 - Take out a sheet of paper
 - Identify the actors within the Scheduler system
 - What are the use cases / what
 - Draw a use case diagram for Scheduler
 - What are the use cases you can identify

Can you do w/ scheduler?

Blackboard

Use Case Types

- 9 • Depend on development phase
 - How far in the project.
- High Level
 - Short one paragraph brief description only
- Essential/Analysis
 - Describe what the system does
 - Should be UI independent
- Real
 - Describe what the system does
 - Focuses on how the UI supports the operations
 - Typically a re-write of the analysis use case

Most detailed.
Detailed always go through first.

Different detail in the use case scenarios.

Use Case Scenario

- Describes
 - describes the context of a use case for a particular user
 - conditions, motivation, and environment of the task for a particular user.
- Used by designers as a way to understand users' motivation and tasks in an interface

why/what

Pieces of a good use case

scenario

- Use case name — *Matches the use case diagram.*
 - The name of the use case
 - Matches the name on a Use Case Diagram
- Actors — *Relationship.*
 - Who is involved in the use case / may use the use case
- High Level Description
 - Describe briefly what the use case does
- Preconditions — *things which must be*
 - Things which must be met before the use case can be executed
- Use Case Flow — *steps.*
 - Describe what happens as the use case flows
- Alternate flows — *Different ways to*
 - What happens if a problem develops *so through the use case.*

Mainly text.

Lets work an example

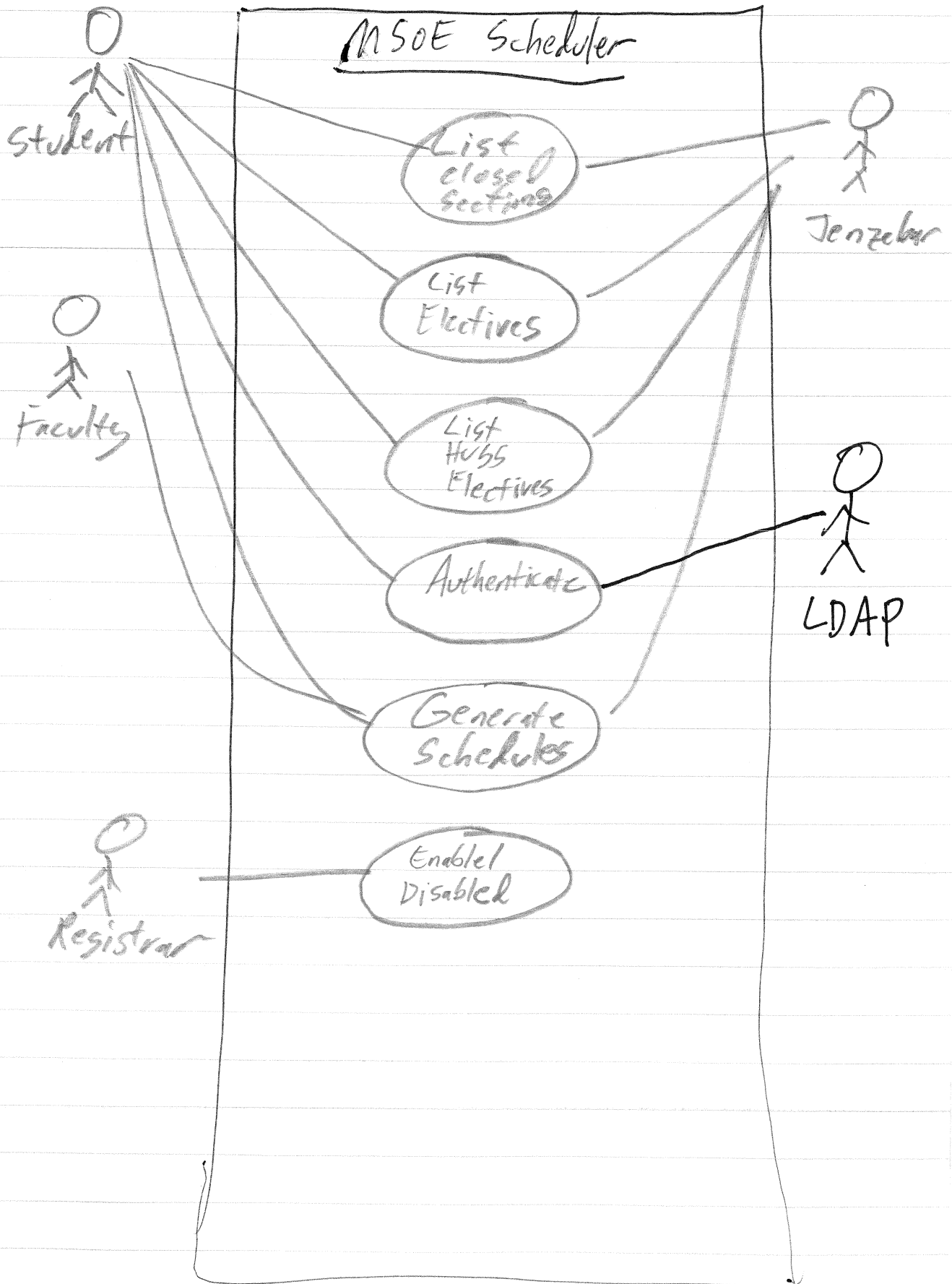
- Scheduler
 - Generate Schedules

Solution – Generate Schedules Use Case

- Actors
 - Student
- High Level Description
 - This use case describes how a student may use the MSOE scheduler to generate potential course schedules. When completed, the student will have a list of potential course schedules that are valid for the given classes the student wishes to take.
- Use Case Flow:
 - 1. Student has a desire to generate a list of schedules and invokes the course scheduler.
 - 2. Students enters a list of courses that they wish to take.
 - 3. Scheduler lists the times at which the desired courses are offered as well as the sections which are closed.
 - 4. The student selects the sections and times which should be considered for schedule generation.
 - 5. The student enters the number of schedules to generate.
 - 6. The student indicates that the system should generate schedules.
 - 7. The system generates a set of schedules and displays them to the user.
 - 8. The user desires a generated schedule and sends it to the printer.
 - 9. The system prints out a schedule to the printer.
 - 10. The user saves the desired schedule to Outlook to import into his / her calendar.
 - 11. The system generates an ics file and saves it to the local computer.
- Alternate flows:
 - 3a1 A student enters a course which is not offered for the given quarter. The system will indicate that the course is not available.
 - 4a1 The student does not select any sections. Scheduler will not generate a schedule for the student.
 - 5a1 The student does not enter a number of schedules to generate. By default, 20 will be generated.
 - 7a1. No valid schedules can be generated. The system will display an error message to the user indicating it is impossible to generate a valid schedule with the desired courses.

Errors/ different paths -

Table of Contents for system,



Name: Generate Schedules

Actor: Student, Faculty, Jenzebar

Preconditions: User has authenticated. ⊆ Not true in existing system
System must be enabled.

High Level Description: This use case allows students to create a set of schedules based on desired courses. When completed, the student will have a set of valid schedules.

Use Case Flow:

1. Student has a desire to generate a list of schedules.
2. Student enters the desired courses.
3. Scheduler shows/displays the section of desired course and the times they are offered.
4. Student selects the sections they would like to consider.
5. Student selects how many schedules he/she would like generated.
6. System generated a set of schedules.