

Week 5: Use Case Modeling and Analysis

Dr. Joshua C. Nwokeji

February 21, 2022

Use Case Diagram

- Describes the interactions between a system and an external actor/users.
- Supports elicitation of user/functional requirements by helping to model and understand software users together with the activities they perform.
- Easy to understand by all stakeholders, especially those with little or no computing background background.
- Can be used in requirements analysis, preparing a test plan, validating a design, etc.
- But not too strong in capturing nonfunctional requirements.

Use Case Jobs Samples

- What types jobs requires competency (skills and knowledge) of use case?
- Thousands of jobs, examples:
 - Business Analyst
 - Business Analyst
 - Requirements Analysts
 - IT Business Process Analyst

Many other Requirements/Business/Systems Analyst Jobs. Find out more at:

- Glassdoor
- Indeed
- Monster

Use Case Elements

- Use cases can be described as a diagram using the following basic elements:
 - Use cases.
 - Actors.
 - Systems Boundary.
 - Relationship or actor-use case communication links
- Other elements in a use case diagram include:
 - Use case extension
 - Use case include
 - Actor and use case generalization.

Use Case Elements: Use Case

- Describes a **standalone activity** performed by an actor, outside the software boundary, to achieve an outcome.
- Use cases are created from userstories: A short description of the activities each user class wants to (or can) perform in a software.
- E.g., *"as a student, I want to view and easily print my syllabus from Blackboard"*.
"As an instructor, I want to upload lecture materials on Blackboard and contact my students", etc.
- Use cases are usually written as **action words** using a **verb-object** format
- Examples: 'print syllabus'; 'upload material'; 'create account'; 'submit assignment', etc.

Use Case Notation

- Use cases are represented as ellipse.
- The name of the use case is written inside the ellipse using the **verb-object** format.
- For instance, see the figure below:

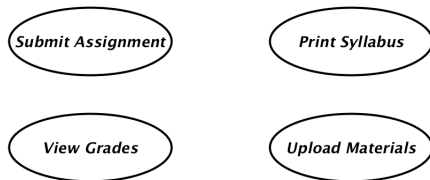


Figure 1: Use Case Notation and Examples of Use Cases

Activity

- Study the description below carefully to identify and draw all use cases.

Gannon University is implementing a *Course Registration Software*. This software allows professors to select the courses they will teach in each semester from a course catalog. Students can enroll in up to 4 courses per semester. The financial status of each student must be confirmed by billing software before the student can enroll. Any student who wants to enroll in more than 4 courses must send a request to the department chair. The chair can approve or deny the request. All users can view, print or download the catalog. Professors can download the course roster to see the number of students enrolled in a class. Students can drop up to 2 classes within the first 3 weeks. Human resource department uses this system to verify academic status before they offer a job to any student.

Activity

- Study the description below carefully to identify and draw all use cases.

Gannon University is implementing a *Course Registration Software*. This software allows professors to **select** the courses they will in teach in each semester from a course catalog. Students can **enroll** in up to 4 courses per semester. The financial status of each student must be **confirmed** by billing software before the student can enroll. Any student who want to enroll in more than 4 courses must **send a request** to the department chair. The chair can **approve or deny the request**. All users can **view, print or download** the catalog. Professors can **download** the course roster to see the number of students enrolled in a class. Students can **drop** up to 2 classes within the first 3 weeks. Human resource department uses this system to **verify** academic status before they offer job to any student.

Use Case Elements: Actor

- Any external entity that interacts with the software.
- An Actor can be:
 - ① Stakeholders such as customers, employees, government agents, etc.
 - ② Another software e.g., a DBMS, automated scaling listener, load balance, etc.
 - ③ External device or hardware e.g., printer, pos, scanner, etc.
 - ④ Organizational units e.g., finance department, hr, etc.

Use Case Elements: Actor Notation

- Usually labeled as name of things/nouns.
- Notation: A stick-like figure with actors name written below the figure. See Figure 2 below

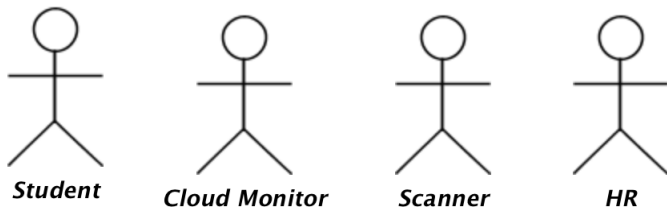


Figure 2: Actor Notation with Examples

Use Case Element: Systems Boundary

- Shows the boundary between the software and external environment.
- Encloses all use cases and separate them from external actors.
- Must be clearly labeled with name of the software usually written at the top.
- Notation: Represented as a large Rectangle, see Figure 3 below:

Systems Boundary Notation

- Notation: Represented as a large rectangle, see Figure 3 below:

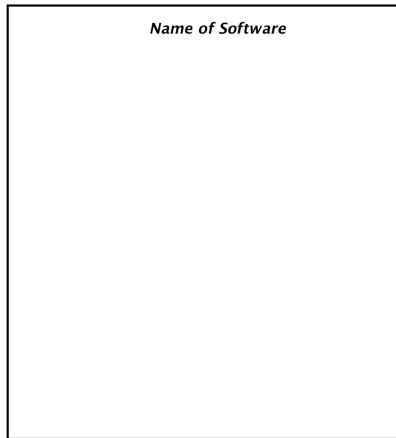


Figure 3: Notation and Examples of Systems Boundary

Actor-Use Case Communication Link

- A line that relates each actor with a use case.
- Usually drawn from the actor to the use case
- Notation: A **solid** line see Figure 4



Figure 4: Use Case-Actor Communication Link Notation

Use Case Diagram: Complete Notation

Use Case Diagram

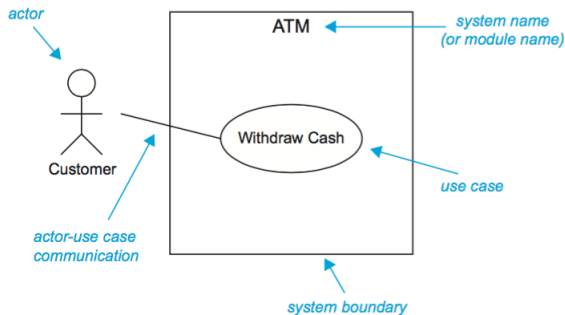


Figure 5: Use Case Diagram Notation

Use Case Example

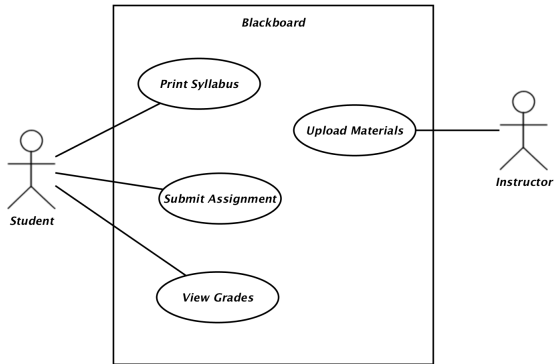


Figure 6: Example of a use case diagram

Activity: Course Registration System

- Draw a use case diagram for the description below

Gannon University is implementing a *Course Registration Software*. This software allows professors to select the courses they will teach in each semester from a course catalog. Students can enroll in up to 4 courses per semester. The financial status of each student must be confirmed by billing software before the student can enroll. Any student who wants to enroll in more than 4 courses must send a request to the department chair. The chair can approve or deny the request. All users can view, print or download the catalog. Professors can download the course roster to see the number of students enrolled in a class. Students can drop up to 2 classes within the first 3 weeks. Human resource department uses this system to verify academic status before they offer a job to any student.

Include Use Case Relationship

- **'Include Use Case Relationship'**: Describes a situation where the behavior of one or more use cases depend on or is inserted in another use case.
- For instance, in order to *check out* in a tilling system, you have to *scan item, calculate amount, make payment*.
- So we can say that *check out* **includes (depends on)** these three uses cases, namely *scan item, calculate amount, and make payment* .
- *Include use cases* can help to:
 - ① Split complex or large use cases (behavior) into simpler ones.
 - ② Extract and re-use common behaviors.
- Can you think of other examples of Include use case?

Notation for Include Use Case Relationship

- Broken line with arrow head.
- The arrow is pointing towards the include use case.
- The term `<<include>>` is written at the center of the line.

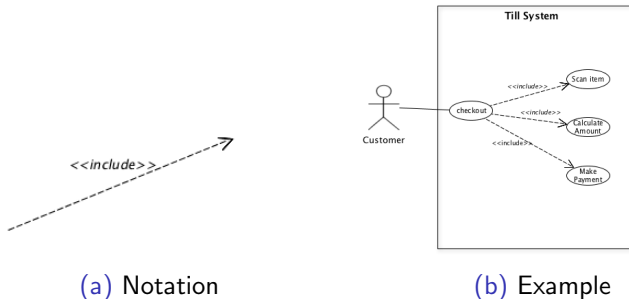


Figure 7: Include Use Case notation and Examples

Include Use Case Contd

- In the previous example, one use case includes or depends on many use cases
- It is possible for many use cases to include or depend on one use case.
- E.g., to *withdraw cash, deposit cash, & transfer funds* in ATM, you have to *enter pin*.

Activity:

- Draw a Use Case Diagram (UCD) for the ATM System.

Include Use Case Contd

- In the previous example, one use case includes or depends on many use cases
- It is possible for many use cases to include or depend on one use case.
- E.g., to *withdraw cash, deposit cash, & transfer funds* in ATM, you have to *enter pin*.

Activity:

- Draw a Use Case Diagram (UCD) for the ATM System.
- Compare this with the UCD in the next slide.

Include Use Case Contd

- In the previous example, one use case includes or depends on many use cases
- It is possible for many use cases to include or depend on one use case.
- E.g., to *withdraw cash, deposit cash, & transfer funds* in ATM, you have to *enter pin*.

Activity:

- Draw a Use Case Diagram (UCD) for the ATM System.
- Compare this with the UCD in the next slide.
- What did you get wrong?

Include UC Example

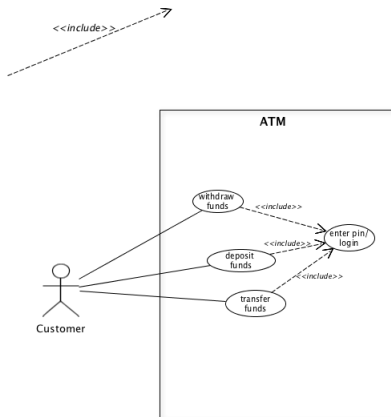
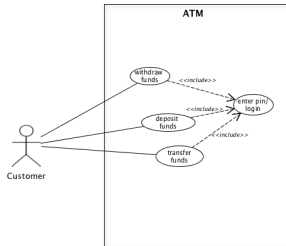
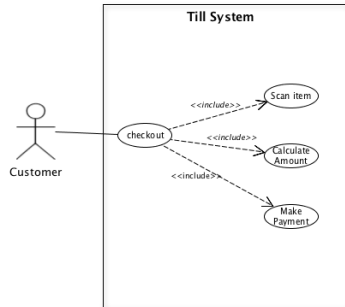


Figure 8: Include Use Case Example

`<<include>>` →



(a) Example 1: Many use cases include 1 use case



(b) Example: One use case include many use cases

Figure 9: Example 2: 1 Use case include many use cases

Use Case Extend Relationship

- The 'extend use case relationship' can be used to specify optional, alternative, or supplementary behaviors (use cases) available for one particular use case (the base use case).
- For instance, in an e-mail system, the base use case '**close email**' has optional use cases such as '**don't save**', '**cancel**' and '**save**'.
- Likewise, the base use case '**make call**' has alternative use cases such as **make conference call** and **record call**
- Extend use cases help us to identify and enhance the system with optional or alternative features to satisfy stakeholder need.

Use Case Extend Relationship Continued

- The extend use cases has a **full functional dependency** on the base use case. For instance, you have to **make a call** first before you can **make a conference call** or **record call**.
- *I.E.*, extend use cases usually cannot function on their own, and may not standalone, without the base use case.
- Extend relationship can be added if you want to specify:
 - Part(s) of a use case which is(are) optional systems behavior.
 - Sub-flow(s) executed after certain conditions are fulfilled.
 - Behavior segments that can be inserted in a base use case.

Notation for Use Case Extend

- Extend relationship is represented with a broken arrow pointing towards the base use case.
- The term <<extension>> is usually written at the center of the line.
- **Observe:** In include relationship the arrow points towards the include use case, but in extend relationship, the arrow points towards the base use case.
- See Figure 10 and 11.

Extend Use Case and Extension Point: Notation

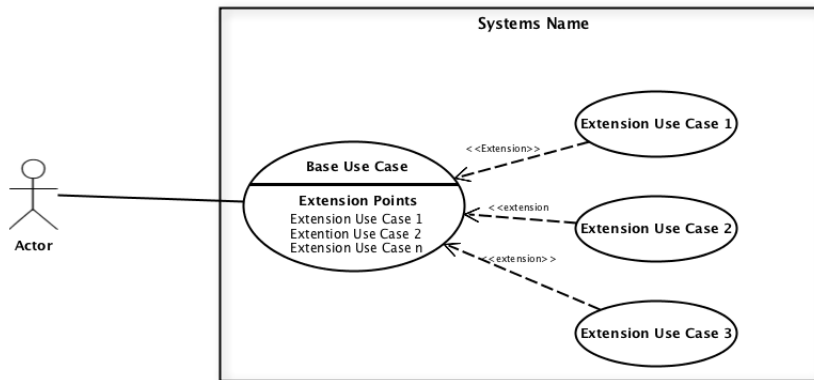


Figure 10: Notation for Extend Use Case and Extension Points

Extend Use Case and Extension Point: Example

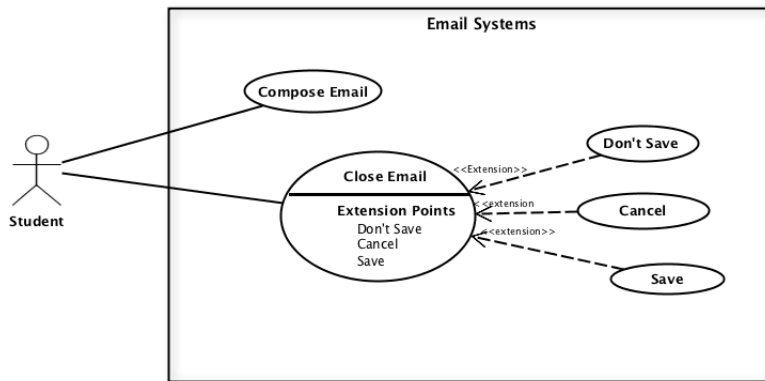


Figure 11: Example of Extend Use Case and Extension Points

Activity

- Draw a use case diagram for the scenario below, showing at least one example of generalization (actor and use case), and include and extend use case relationship.

Pennsylvania State Electoral Commission wants to develop a software to facilitate state elections. In order to cast vote, existing voters must login with usernames and passwords, while new voters must create account before they can login. After logging in, voters can view profile of each candidate, review their manifestos, before selecting the candidate to vote for. Once a candidate is selected, voters have the choice to confirm candidate or cancel voting process or save voting process. Once the voting is completed voters can log out of the system. After the election, electoral officer will collate the result, calculate the number of votes for each candidate, and publish the results. Candidate can check their results 24 hours after the election. If they are not happy with the election outcome, Candidates can complete the complaints form and send it to the electoral commission.

Limitation of Use Case Diagram

- Use case diagram is a good starting point for understanding user and functional requirements.
- But it lacks some more information that can be useful to developers and other stakeholders.
- Useful details that can support in-depth analysis are not included in use case diagram.
- Hence, use case diagrams are usually completed with a **use description**.

Use Case Description

- A typical use case description extends use diagram with the following elements:
 - Use Case ID
 - Use Case Name
 - Description
 - Priority
 - Trigger
 - Pre Conditions
 - Post Conditions
 - Normal Course or Flow
 - Exceptions
 - Alternative Course
- These information are usually captured with a **use case description template**
- Some use case description template can contain other relevant information such as input and output data, business rule, assumption, etc.

Elements of a Use Case Description

| | |
|----------------------------|---|
| Use Case ID | <i>Provides a unique code for identifying each use case. Can be alphanumeric e.g., uc001 or just numeric e.g., 001.</i> |
| Use Case Name | <i>State the name of each use case. Written in verb-object format e.g., Send notification, create account.</i> |
| Description (Goal/Outcome) | <i>Provides a brief explanation of each use case including the intended goal or outcome. E.g., this use case allows any user to create account using personal details including name, email, phone, username and password.</i> |
| Priority | <i>State the priority of the use case e.g., high, medium, low.</i> |
| Trigger | <i>State the event or circumstance that initiated the use case. E.g., payment is due;</i> |
| Actor | <i>Outlines the external entities involved in the use case</i> |
| Pre-conditions | <i>Describes the State of the system before executing the use Case or conditions that must hold or be true for the Use Case to execute. E.g., a student must login to view grade;</i> |
| Post-conditions | <i>Describes the state of the system after the use case has been executed, or certain conditions that must be true after executing the use case. E.g., confirmation email must be sent after payment. User details must be stored in the database after an account is created.</i> |
| Normal Flow | <i>Provides a detailed and step by step flow of events for successful completion of a use case. Shows user activity and systems response</i> <i>E.g., Withdraw cash</i> 1. User inserts card 2. System prompts for pin 3. User enters pin 4. System validates pin 5. System prompts for amount 6. User enters amounts 7. System issues amount 8. System prompts user to collect card 9. User collects card <i>Use case ends.</i> |
| Exceptions | <i>List any condition that can lead to failure of the Use Case</i> <i>E.g., 4. Invalid pin</i> 4a. System prompts the user to re-enter pin 4b. User re-enters pin 4c. System validates pin |
| Alternative flow | <i>List any variations to the main flow or MSS</i> <i>E.g., assuming ATM allows login with Thump Print, then alternative course would be:</i> 1. User insert card 2. System prompt for thump print 3. User provides thump print 4. System validates thump print 5. System prompts for amount 6. User enters amount 7. System issues amount 8. System prompt user to collect card 9. User collects card <i>Use case ends.</i> |

Example of a Use Case Description

| | |
|--------------------|---|
| ID and Name: | UC-4 Request a Chemical |
| Created By: | Lori |
| Date Created: | 8/22/13 |
| Primary Actor: | Requester |
| Secondary Actors: | Buyer, Chemical Stockroom, Training Database |
| Description: | The Requester specifies the desired chemical to request by entering its name or chemical ID number or by importing its structure from a chemical drawing tool. The system either offers the Requester a container of the chemical from the chemical stockroom or lets the Requester order one from a vendor. |
| Trigger: | Requester indicates that he wants to request a chemical. |
| Preconditions: | PRE-1. User's identity has been authenticated. PRE-2. User is authorized to request chemicals. PRE-3. Chemical inventory database is online. |
| Postconditions: | POST-1. Request is stored in the CTS. POST-2. Request was sent to the Chemical Stockroom or to a Buyer. |
| Normal Flow: | 4.0 Request a Chemical from the Chemical Stockroom 1. Requester specifies the desired chemical. 2. System lists containers of the desired chemical that are in the chemical stockroom, if any. 3. System gives Requester the option to View Container History for any container. 4. Requester selects a specific container or asks to place a vendor order (see 4.1). 5. Requester enters other information to complete the request. 6. System stores the request and notifies the Chemical Stockroom. |
| Alternative Flows: | 4.1 Request a Chemical from a Vendor 1. Requester searches vendor catalogs for the chemical (see 4.1.E1). 2. System displays a list of vendors for the chemical with available container sizes, grades, and prices. 3. Requester selects a vendor, container size, grade, and number of containers. 4. Requester enters other information to complete the request. 5. System stores the request and notifies the Buyer. |
| Exceptions: | 4.1.E1 Chemical Is Not Commercially Available 1. System displays message: No vendors for that chemical. 2. System asks Requester if he wants to request another chemical (3a) or to exit (4a). 3a. Requester asks to request another chemical. 3b. System starts normal flow over. 4a. Requester asks to exit. 4b. System terminates use case. |
| Priority: | High |
| Frequency of Use: | Approximately 5 times per week by each chemist, 200 times per week by chemical stockroom staff |
| Business Rules: | BR-28, BR-31 |
| Other Information: | The system must be able to import a chemical structure in the standard encoded form from any of the supported chemical drawing packages. |
| Assumptions: | Imported chemical structures are assumed to be valid. |

Activity

Using the template in the figure below, describe the use case: 'submit assignment'.

| | |
|----------------------------|--|
| Use Case ID | Provides a unique code for identifying each use case. Can be alphanumeric e.g., uc001 or just numeric e.g., 001. |
| Use Case Name | State the name of each use case. Written in verb-object format e.g., Send notification, create account. |
| Description (Goal/Outcome) | Provides a brief explanation of each use case including the intended goal or outcome. E.g., this use case allows any user to create account using personal details including name, email, phone, username and password. |
| Priority | State the priority of the use case e.g., high, medium, low. |
| Trigger | State the event or circumstance that initiated the use case. E.g., payment is due. |
| Actor | Outlines the external entities involved in the use case. |
| Pre-conditions | Describes the State of the system before executing the use Case or conditions that must hold or be true for the Use Case to execute. E.g., a student must login to view grade; |
| Post-conditions | Describes the state of the system after the use case has been executed, or certain conditions that must be true after executing the use case. E.g., confirmation email must be sent after payment. User details must be stored in the database after an account is created. |
| Normal Flow | Provides a detailed and step by step flow of events for successful completion of a use case. Shows user activity and systems response E.g., Withdraw cash 1. User inserts card 2. System prompts for pin 3. User enters pin 4. System validates pin 5. System prompts for amount 6. User enters amounts 7. System issues amount 8. System prompts user to collect card 9. User collects card Use case ends. |
| Exceptions | List any condition that can lead to failure of the Use Case E.g., 4. Invalid pin 4a. System prompts the user to re-enter pin 4b. User re-enters pin 4c. System validates pin |
| Alternative flow | List any variations to the main flow or MSS E.g., assuming ATM allows login with Thump Print, then alternative course would be: 1. User inserts card 2. System prompt for thump print 3. User provides thump print 4. System validates thump print 5. System prompts for amount 6. User enters amount 7. System issues amount 8. System prompt user to collect card 9. User collects card Use case ends. |

Announcement

- Any questions on Written Report 1–Project Vision and Scope 2 .
- Due Date: Monday, February 21, 2021 by 11.59pm
- Submit via Blackboard.