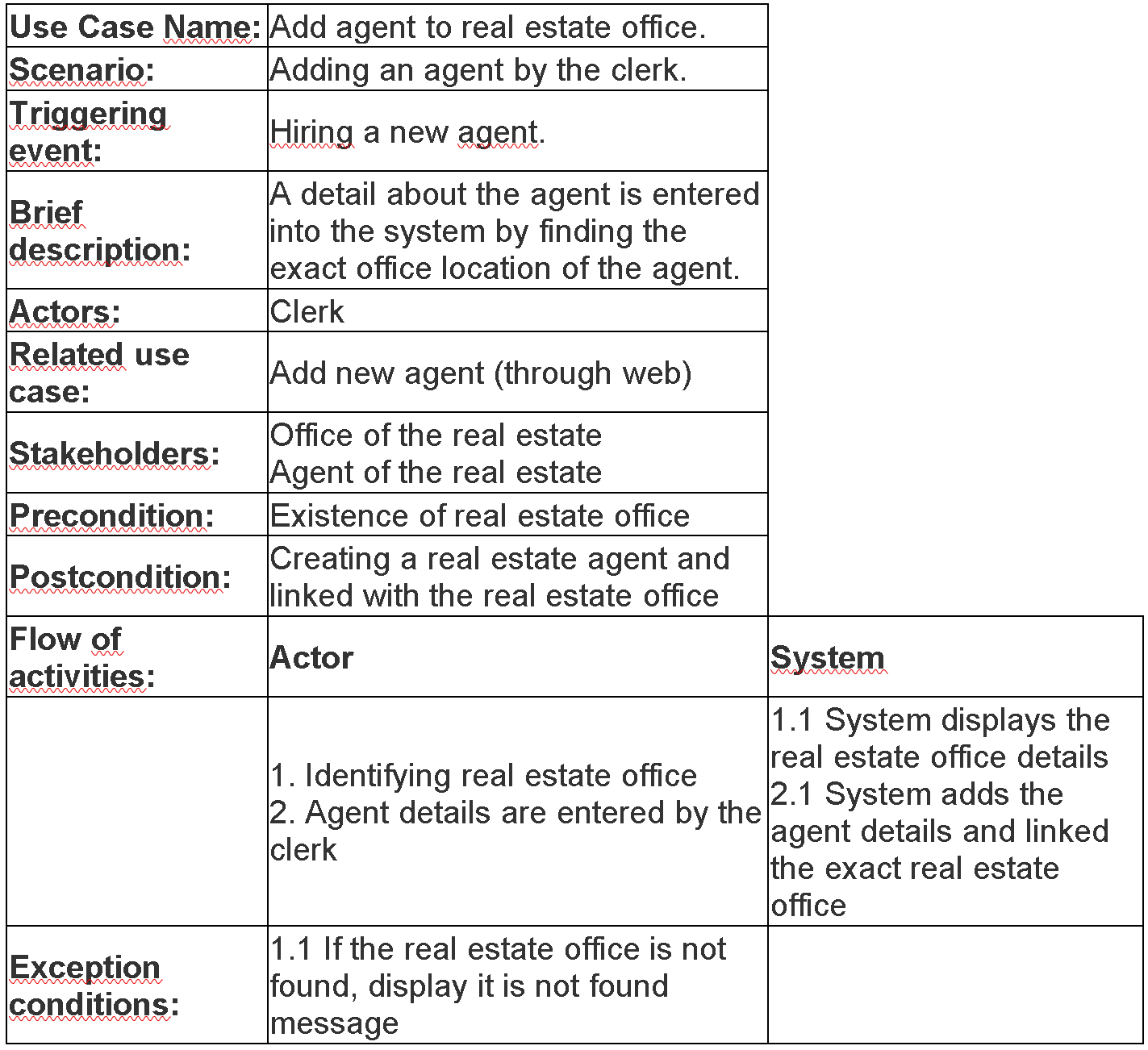
***PORTFOLIO OF EXERCISES***

***Use Case Narratives Exercises:***

***Exercise 1*** 🡪 <https://www.chegg.com/homework-help/systems-analysis-and-design-in-a-changing-world-6th-edition-chapter-5-solutions-9781133713371>

<https://mygust.com/uploads/BOOK-Systems_analysis_and_design_in_a_changin.pdf>

******

**PROMPT:** The following description includes many use cases: "One of the functions of the Board of Realtors is to provide a Multiple Listing Service (MLS) system that supplies information that local real estate agents use to help them sell houses to their customers. During the month, agents list houses for sale (listings) by contracting with homeowners. The agent works for a real estate office, which sends information on the listing to the MLS. Therefore, any agent in the community can get information on the listing. Information on a listing includes the address, year built, square feet, number of bedrooms, number of bathrooms, owner name, owner phone number, asking price, and status code. At any time during the month, an agent might directly request information on listings that match customer requirements, so the agent contacts the MLS with the request. Information is provided on the house, on the agent who listed the house, and on the real estate office for which the agent works. For example, an agent might want to call the listing agent to ask additional questions or call the homeowner directly to make an appointment to show the house. Once each week, the MLS produces a listing book that contains information on all listings. These books are sent to some real estate agents. Some real estate agents want the books (which are easier to flip through), so they are provided even though the information is often out of date. Sometimes, agents and owners decide to change information about a listing, such as reducing the price, correcting previous information on the house, or indicating that the house is sold. The real estate office sends in these change requests to the MLS when the agent asks the office to do so." One of the use cases identified in the text is "Add agent to real estate office". I need you to analyze the text and to write an appropriate Use Case description of the use case "Add agent to real estate office". Please take into account the following information when writing the use case description: • “Use Case Name” is any name that is the simplest possible but descriptive too. • Each use case contains the internal activities set which are unique and it is represented as “Scenario”. • “Triggering event” is basically an event that is initialized by the primary actor to start use case execution. • “Brief description” is to describe the whole use case description in a short form. • “Actor” can be a human user, a hardware device, or a software system that interacts with the system for goal accomplishment. • “Related use case” represents the associated use cases. • “Stakeholders” are the persons who are all involved in the successful execution of the system. • “Precondition” is a condition that should be met before the use case starts. • “Postcondition” is a condition that should be met after the use case completed successfully. • “Flow of activities” is the tenth compartment of the use case description. o It consists of two columns about the steps performed by the actor and the response of the system. • “Exception conditions” describes the details of the other activities and the exception conditions.

ChatGPT

**ChatGPT**

**Use Case Name:** Add Agent to Real Estate Office

**Triggering Event:** Real estate office decides to add a new agent to their team.

**Brief Description:** This use case involves the process of adding a new agent to a real estate office within the context of the Board of Realtors system.

**Actor:** Real Estate Office Manager/Administrator

**Related Use Case:** None directly related.

**Stakeholders:**

1. Real Estate Office Manager/Administrator: Initiates the process to add a new agent.
2. Board of Realtors System: Manages and records information related to real estate agents and listings.

**Precondition:** The real estate office is registered with the Board of Realtors system, and the system is operational.

**Postcondition:** The new agent is successfully added to the real estate office, and their information is updated in the Board of Realtors system.

**Flow of Activities:**

| **Actor's Steps** | **System's Response** |
| --- | --- |
| 1. Real estate office manager logs into the Board of Realtors system. | 2. System validates the manager's credentials and grants access. |
| 3. Manager selects the option to add a new agent to the office. | 4. System prompts the manager to enter the new agent's details (name, contact information, etc.). |
| 5. Manager enters the required information and confirms the addition. | 6. System validates the data and updates the real estate office's records. |
| 7. System notifies the manager of the successful addition of the new agent. | 8. Manager receives a confirmation message and can view the updated list of agents. |

**Exception Conditions:**

* If the manager's credentials are invalid, the system denies access and informs the manager to provide correct credentials.
* If there is a technical issue preventing the addition of the new agent, the system displays an error message, and the manager is instructed to try again later.
* If the entered information for the new agent is incomplete or incorrect, the system prompts the manager to correct the details before proceeding.
* In case the real estate office is not registered with the Board of Realtors system, the manager is informed that registration is required before adding a new agent.

This use case ensures a systematic and secure process for adding new agents to a real estate office, maintaining accurate records within the Board of Realtors system.

**Analysis:**

In comparing the two solutions for the "Add Agent to Real Estate Office" use case, both share common elements in terms of the triggering event, actors, stakeholders, preconditions, postconditions, and the core flow of activities. However, there are notable distinctions in their approach. The original solution offers a more detailed brief description, emphasizing the clerk's role in entering agent details and specifying the process of finding the exact office location. In contrast, the revised solution opts for brevity, omitting an explicit mention of the brief description. The original solution also lacks a related use case, while the revised version introduces the idea of an alternative method for adding a new agent through the web.

Moreover, the flow of activities in the original solution includes more detailed substeps, providing a granular view of the clerk's actions and the system's responses. On the other hand, the revised solution simplifies the flow, presenting concise steps for identification and data entry. Both solutions maintain consistency in identifying stakeholders and ensuring the existence of the real estate office as a precondition.

In summary, the original solution excels in providing detailed steps and a nuanced brief description, while the revised solution prioritizes brevity and introduces the concept of a related use case. Combining the strengths of both approaches could result in a comprehensive and concise use case description.

***EXERCISE 2🡪*** <https://www.chegg.com/homework-help/systems-analysis-and-design-in-a-changing-world-6th-edition-chapter-5-problem-2cbr-solution-9781133713371>

<https://mygust.com/uploads/BOOK-Systems_analysis_and_design_in_a_changin.pdf>

| **Use Case Name:** | Create new listing |
| --- | --- |
| **Scenario:** | Creating new list by the clerk |
| **Triggering event:** | Getting a new property to sale |
| **Brief description:** | Identify and verify the agent who created the listing. Then add the details of the property to the system |
| **Actors:** | Clerk |
| **Related use case:** | Agent/Office creates the listing (through web). |
| **Stakeholders:** | Office of the real estate  Agent of the real estate  Owner of the property |
| **Preconditions:** | Existence of real estate office.  Existence of real estate agent |
| **Postconditions:** | Creating a listing and linked with the real estate office and real estate agent |
| **Flow of activities:** | **Actor** | **System** |
|  | 1. Identifying real estate agent  2. Listing details are entered by the clerk | 1.1 System displays the real estate agent and corresponding office details  2.1 System creates listing and links the exact real estate agent and then displays the listing details |
| **Exception conditions:** | 1.1 If the real estate agent details are not found, display the message “Not found”. |  |

**PROMPT:** The following description includes many use cases: "One of the functions of the Board of Realtors is to provide a Multiple Listing Service (MLS) system that supplies information that local real estate agents use to help them sell houses to their customers. During the month, agents list houses for sale (listings) by contracting with homeowners. The agent works for a real estate office, which sends information on the listing to the MLS. Therefore, any agent in the community can get information on the listing. Information on a listing includes the address, year built, square feet, number of bedrooms, number of bathrooms, owner name, owner phone number, asking price, and status code. At any time during the month, an agent might directly request information on listings that match customer requirements, so the agent contacts the MLS with the request. Information is provided on the house, on the agent who listed the house, and on the real estate office for which the agent works. For example, an agent might want to call the listing agent to ask additional questions or call the homeowner directly to make an appointment to show the house. Once each week, the MLS produces a listing book that contains information on all listings. These books are sent to some real estate agents. Some real estate agents want the books (which are easier to flip through), so they are provided even though the information is often out of date. Sometimes, agents and owners decide to change information about a listing, such as reducing the price, correcting previous information on the house, or indicating that the house is sold. The real estate office sends in these change requests to the MLS when the agent asks the office to do so." One of the use cases identified in the text is "create new listing". I need you to analyze the text and to write an appropriate Use Case description of the use case "create a new listing". Please take into account the following information when writing the use case description: • “Use Case Name” is any name that is the simplest possible but descriptive too. • "Scenario": Each use case contains the internal activities set which are unique and it is represented as a scenario . • “Triggering event” is basically an event that is initialized by the primary actor to start use case execution. • “Brief description”: is to describe the whole use case description in a short form. • “Actor” can be a human user, a hardware device, or a software system that interacts with the system for goal accomplishment. • “Related use case” represents the associated use cases. • “Stakeholders” are the persons who are all involved in the successful execution of the system. • “Precondition” is a condition that should be met before the use case starts. • “Postcondition” is a condition that should be met after the use case completed successfully. • “Flow of activities” is the tenth compartment of the use case description. o It consists of two columns about the steps performed by the actor and the response of the system. • “Exception conditions” describes the details of the other activities and the exception conditions.

ChatGPT

**ChatGPT**

**Use Case Name:** Create New Listing

**Scenario:** Creating a new listing for a property.

**Triggering Event:** A real estate agent initiates the process of listing a property for sale.

**Brief Description:** This use case involves the process of adding a new property listing to the Multiple Listing Service (MLS) system, enabling real estate agents to access and share information about the property with potential buyers.

**Actor:** Real Estate Agent

**Related Use Case:** None directly related.

**Stakeholders:**

1. Real Estate Agent: Initiates the process of creating a new listing.
2. Homeowner: Provides information about the property to be listed.
3. Multiple Listing Service (MLS) System: Records and disseminates information about property listings.

**Precondition:** The real estate agent is registered with the MLS system, and the system is operational.

**Postcondition:** The new property listing is successfully added to the MLS system and is accessible to other real estate agents.

**Flow of Activities:**

| **Actor's Steps** | **System's Response** |
| --- | --- |
| 1. Real estate agent logs into the MLS system. | 2. System validates the agent's credentials and grants access. |
| 3. Agent selects the option to create a new listing. | 4. System prompts the agent to enter details about the property (e.g., address, year built, square footage, bedrooms, bathrooms, owner's contact information, asking price). |
| 5. Agent enters the required information and submits the listing. | 6. System validates the data and adds the new listing to the MLS database. |
| 7. System confirms successful creation of the listing. | 8. Agent receives a confirmation message and can view the newly created listing in the MLS system. |

**Exception Conditions:**

* If the agent's credentials are invalid, the system denies access and prompts the agent to provide correct credentials.
* If there are technical issues preventing the creation of the listing, the system displays an error message and instructs the agent to try again later.
* If the entered information for the new listing is incomplete or incorrect, the system prompts the agent to correct the details before submission.
* In case the MLS system is unavailable, the agent is informed of the temporary outage and advised to try again when the system is back online.

**Analysis:**

The provided solution for the "Create New Listing" use case adopts a similar approach to the revised solution generated, emphasizing the role of the clerk in creating a new property listing. Both solutions share common elements such as the triggering event of acquiring a new property for sale, involvement of the clerk as the actor, and the importance of identifying and verifying the agent responsible for the listing. Additionally, both solutions recognize the stakeholders, including the real estate office, real estate agent, and property owner, and emphasize preconditions like the existence of the real estate office and agent.

Differences lie in the level of detail and structure. The provided solution opts for a more succinct brief description and a straightforward flow of activities, outlining the identification of the real estate agent and the entry of listing details by the clerk. In contrast, the revised solution provides a bit more detail in the flow of activities, breaking down the steps performed by the actor and the system's responses. Additionally, the generated solution introduces a related use case involving web-based listing creation, offering an alternative method for agents.

Overall, both solutions effectively capture the essence of the "Create New Listing" use case, with variations in the level of detail and structure chosen for the description. The provided solution maintains simplicity and clarity, while my revised solution leans towards a more comprehensive breakdown of activities and considerations.

***EXERCISE 3🡪*** <https://www.chegg.com/homework-help/systems-analysis-and-design-in-a-changing-world-6th-edition-chapter-5-problem-1osc-solution-9781133713371> <https://mygust.com/uploads/BOOK-Systems_analysis_and_design_in_a_changin.pdf>

| **Use Case Name:** | Request a package pickup |
| --- | --- |
| **Scenario:** | Requesting a package pickup through online. |
| **Triggering event:** | Customer request for a package pickup through online |
| **Brief description:** | Customer wants the courier service to pick up the package by requesting through online.  The package details such as delivery address, service type, size and weight of the package.  The system calculates the cost, expected time of pickup and prints the delivery label. |
| **Actors:** | Customer |
| **Related use case:** | Customer request for a package pickup through phone  Customer request for a package pickup in-person |
| **Stakeholders:** | Customer  Employee  Bill |
| **Preconditions:** | Account for customer should exit |
| **Post conditions:** | Creating a pickup request and link with the customer account |
| **Flow of activities:** | **Actor** | **System** |
|  | 1. For every package  1.1 The delivery address is entered by the customer  1.2 The service type is entered by the customer  1.3 The size and weight of the package is entered by the customer  1.4 Customer requests the delivery label | 1.3.1System calculates the cost  1.3.2 System shows the expected time of pickup  1.3.3 System creates the record for request and package  1.4.1System prints the delivery label |
| **Exception conditions:** | 1.4. Failure of printing the label |  |

**PROMPT:** The following description includes many use cases: "As On the Spot Courier Services continues to grow, Bill discovers that he can provide much better services to his customers if he utilizes some of the technology that is currently available. For example, it will allow him to maintain frequent communication with his delivery trucks, which could save transportation and labor costs by making the pickup and delivery operations more efficient. This would allow him to serve his customers better. Of course, a more sophisticated system will be needed, but Bill’s development consultant has assured him that a straightforward and not too complex solution can be developed. Here is how Bill wants his business to operate. Each truck will have a morning and an afternoon delivery and pickup run. Each driver will have a portable digital device with a touch screen. The driver will be able to view his or her scheduled pickups and deliveries for that run. (Note: This process will require a new use case—something the Agile development methodology predicted would happen.) However, because the trucks will maintain frequent contact with the home office via telephony Internet access, the pickup/delivery schedule can be updated in real-time—even during a run. Rather than maintain constant contact, Bill decides that it will be sufficient if the digital device synchronizes with the home office whenever a pickup or delivery is made. At those points in time, the route schedule can be updated with appropriate information. Previously, customers were able to either call On the Spot and request a package pickup or visit the company’s website to schedule a pickup. Once customers logged in, they could go to a webpage that allowed them to enter information about each package, including “deliver to” addresses, size and weight category information, and type of service requested. On the Spot provided “three-hour,” “same day,” and “overnight” services. To facilitate customer self-service, On the Spot didn’t require exact weights and sizes, but there were predefined size and weight categories from which the customer could choose. Once the customer entered the information for all the packages, the system would calculate the cost and then print mailing labels and receipts. Depending on the type of service requested and the proximity of a delivery truck, the system would schedule an immediate pickup or one for later that day. It would display this information so the customer would immediately know when to expect the pickup. Picking up packages was a fairly straightforward process. But there was some variation in what would happen depending on what information was in the system and whether the packages were already labeled. Upon arriving at the scheduled pickup location, the driver would have the system display any package information available for this customer. If the system already had information on the packages, the driver would simply verify that the correct information was already in the system for the packages. The driver could also make such changes as correcting the address, deleting packages, or adding new packages. If this were a cash customer, the driver would collect any money and enter that into the system. Using a portable printer from the van, the driver could print a receipt for the customer as necessary. If there were new packages that weren’t in the system, the driver would enter the required information and also print mailing labels with his portable printer. One other service that customers required was to be able to track the delivery status of their packages. The system needed to track the status of a package from the first time it “knew” about the package until it was delivered. Such statuses as “ready for pickup,” “picked up,” “arrived at warehouse,” “out for delivery,” and “delivered” were important. Usually, a package would follow through all the statuses, but due to the sophistication of the scheduling and delivery algorithm, a package would sometimes be picked up and delivered on the same delivery run. Bill also decided to add a status of “cancelled” for those packages that were scheduled to be picked up but ended up not being sent." One of the use cases identified in the text is "request a package pick-up". I need you to analyze the text and to write an appropriate Use Case description of the use case "request a package pick-up". Please take into account the following information when writing the use case description: • “Use Case Name” is any name that is the simplest possible but descriptive too. • "Scenario": Each use case contains the internal activities set which are unique and it is represented as a scenario . • “Triggering event” is basically an event that is initialized by the primary actor to start use case execution. • “Brief description”: is to describe the whole use case description in a short form. • “Actor” can be a human user, a hardware device, or a software system that interacts with the system for goal accomplishment. • “Related use case” represents the associated use cases. • “Stakeholders” are the persons who are all involved in the successful execution of the system. • “Precondition” is a condition that should be met before the use case starts. • “Postcondition” is a condition that should be met after the use case completed successfully. • “Flow of activities” is the tenth compartment of the use case description. o It consists of two columns about the steps performed by the actor and the response of the system. • “Exception conditions” describes the details of the other activities and the exception conditions.

ChatGPT

**ChatGPT**

**Use Case Name:** Request Package Pick-Up

**Scenario:** Initiating a package pick-up request by a customer.

**Triggering Event:** A customer initiates a request for On the Spot Courier Services to pick up a package.

**Brief Description:** This use case involves customers requesting the pick-up of packages through On the Spot Courier Services. Customers can log in to the system via the company's website, provide package details, select service options, and schedule a pick-up.

**Actor:** Customer

**Related Use Case:** None directly related.

**Stakeholders:**

1. Customer: Initiates the request for package pick-up.
2. On the Spot Courier Services: Manages and fulfills pick-up requests.

**Precondition:** The customer is registered on the On the Spot Courier Services website, and the system is operational.

**Postcondition:** The system acknowledges the pick-up request, schedules it based on the selected service, and updates the customer on the expected pick-up time.

**Flow of Activities:**

| **Actor's Steps** | **System's Response** |
| --- | --- |
| 1. Customer logs in to the company's website. | 2. System validates customer credentials and grants access. |
| 3. Customer navigates to the pick-up request page. | 4. System displays a webpage allowing the customer to enter package details (size, weight, "deliver to" address) and select service options ("three-hour," "same day," or "overnight"). |
| 5. Customer submits package information and service preferences. | 6. System calculates the cost, schedules the pick-up based on service type and proximity of delivery truck, and displays the scheduled pick-up time to the customer. |
| 7. Customer reviews the scheduled pick-up details and confirms the request. | 8. System confirms the pick-up request, generates and prints mailing labels, and updates the package status to "ready for pickup." |

**Exception Conditions:**

* If the customer's login credentials are invalid, the system denies access and prompts the customer to provide correct credentials.
* If there are technical issues preventing the submission of package details, the system displays an error message and instructs the customer to try again later.
* If the entered information for the pick-up request is incomplete or incorrect, the system prompts the customer to correct the details before submission.
* In case the system cannot schedule an immediate pick-up due to operational constraints, it informs the customer of the next available slot and confirms the scheduled pick-up time.

This use case ensures a seamless process for customers to request package pick-ups through the On the Spot Courier Services website, providing flexibility in service options and real-time scheduling information.

**Analysis:**

The provided solution for the "Request Package Pick-Up" use case aligns closely with the generated solution, sharing similarities in the overall structure and key components. Both solutions recognize the essential steps involved in a customer initiating a pick-up request online, including entering package details, calculating costs, displaying expected pickup times, and printing delivery labels.

Differences lie in the level of detail and focus. The provided solution offers a concise and straightforward flow of activities, emphasizing the customer's actions and the corresponding system responses. In contrast, the solution generated by ChatGPT breaks down the steps into sub activities, providing a more granular view of the process. Additionally, the provided solution includes related use cases for requesting a package pickup through phone or in-person, which adds completeness to the use case description. The narrative generated by the AI, however, opted for simplicity by not introducing related use cases.

Ultimately, both solutions effectively capture the essence of the use case, with variations in detail and emphasis. The provided solution highlights related use cases, while the generated solution offers a slightly more detailed breakdown of activities. The choice between the two depends on the desired level of detail and the need for related use case inclusion.

***EXERCISE 4:***🡪(EXERCISE 4 form UCD) <https://cmps-people.ok.ubc.ca/bowenhui/310/8-UML.pdf>

Title: Withdraw money using a Visa card

Summary: this use case allows a Visa card holder, who is not a customer of the bank, to withdraw money if his or her daily limit allows it.

Actors: Visa CardHolder (primary), Visa AS (secondary).

Flow of events

Preconditions:

•The ATM cash box is well stocked.

•There is no card in the reader.

Main success scenario:

1. The Visa CardHolder inserts his or her smartcard in the ATM’s card reader.

2. The ATM verifies that the card that has been inserted is indeed a smartcard.

3. The ATM asks the Visa CardHolder to enter his or her pin number.

4. The Visa CardHolder enters his or her pin number.

5. The ATM compares the pin number with the one that is encoded on the chip of the smartcard.

6. The ATM requests an authorisation from the VISA authorisation system.

7. The VISA authorisation system confirms its agreement and indicates the daily withdrawal limit.

8. The ATM asks the Visa CardHolder to enter the desired withdrawal amount.

9. The Visa CardHolder enters the desired withdrawal amount.

10. The ATM checks the desired amount against the daily withdrawal limit.

11. The ATM asks the Visa CardHolder if he or she would like a receipt.

12. The Visa CardHolder requests a receipt.

13. The ATM returns the card to the Visa CardHolder.

14. The Visa CardHolder takes his or her card.

15. The ATM issues the banknotes and a receipt.

16. The Visa CardHolder takes the banknotes and the receipt.

“Alternative” sequences:

A1: temporarily incorrect pin number

The A1 sequence starts at point 5 of the main success scenario.

6. The ATM informs the CardHolder that the pin is incorrect for the first or second time.

7. The ATM records the failure on the smartcard.

The scenario goes back to point 3.

A2: the amount requested is greater than the daily withdrawal limit

The A2 sequence starts at point 10 of the main success scenario.

11. The ATM informs the CardHolder that the amount requested is greater than the daily withdrawal limit.

The scenario goes back to point 8.

A3: the Visa CardHolder does not want a receipt

The A3 sequence starts at point 11 of the main success scenario.

12. The Visa CardHolder declines the offer of a receipt.

13. The ATM returns the smartcard to the Visa CardHolder.

14. The Visa CardHolder takes his or her smartcard.

15. The ATM issues the banknotes.

16. The Visa CardHolder takes the banknotes.

Error sequences:

E1: invalid card

The E1 sequence starts at point 2 of the main success scenario.

1. The ATM informs the Visa CardHolder that the smartcard is not valid (unreadable, expired, etc.) and confiscates it; the use case fails.

E2: conclusively incorrect pin number

The E2 sequence starts at point 5 of the main success scenario.

6. The ATM informs the Visa CardHolder that the pin is incorrect for the third time.

7. The ATM confiscates the smartcard.

8. The VISA authorisation system is notified; the use case fails.

E3: unauthorised withdrawal

The E3 sequence starts at point 6 of the main success scenario.

7. The VISA authorisation system forbids any withdrawal.

8. The ATM ejects the smartcard; the use case fails.

E4: the card is not taken back by the holder

The E4 sequence starts at point 13 of the main success scenario.

14. After 15 seconds, the ATM confiscates the smartcard.

15. The VISA authorisation system is notified; the use case fails.

E5: the banknotes are not taken by the holder

The E5 sequence starts at point 15 of the main success scenario.

16. After 30 seconds, the ATM takes back the banknotes.

17. The VISA authorisation system is informed; the use case fails

Postconditions:

•The cashbox of the ATM contains fewer notes than it did at the start of the use case (the number of notes missing depends on the withdrawal amount).

**PROMPT:** The following description includes many use cases: "This case study concerns a simplified system of the automatic teller machine (ATM). The ATM offers the following services: 1. Distribution of money to every holder of a smartcard via a card reader and a cash dispenser. 2. Consultation of account balance, cash and cheque deposit facilities for bank customers who hold a smartcard from their bank. Do not forget either that: 3. All transactions are made secure. 4. It is sometimes necessary to refill the dispenser, etc. Let’s look at each of the sentences of the exposition in turn. Sentence 1 allows us to identify an obvious initial actor straight away: every “holder of a smartcard”. He or she will be able to use the ATM to withdraw money using his or her smartcard. However, be careful: the card reader and cash dispenser constitute part of the ATM. They can therefore not be considered as actors! You can note down that the identification of actors requires the boundary between the system being studied and its environment to be set out exactly. If we restrict the study to the control/ command system of physical elements of the ATM, the card reader and cash dispenser then become actors. Another trap: is the smartcard itself an actor? The card is certainly external to the ATM, and it interacts with it... Yet, we do not recommend that you list it as an actor, as we are putting into practice the following principle: eliminate “physical” actors as much as possible to the advantage of “logical” actors. The actor is the who or what those benefits from using the system. It is the card holder who withdraws money to spend it, not the card itself! Sentence 2 identifies additional services that are only offered to bank customers who hold a smartcard from this bank. This is therefore a different profile from the previous one, which we will realize by a second actor called Bank customer. Sentence 3 encourages us to take into account the fact that all transactions are made secure. But who makes them secure? There are therefore other external entities, which play the role of authorization system and with which the ATM communicates directly. An interview with the domain expert4 is necessary to allow us to identify two different actors: • the Visa authorization system (VISA AS) for withdrawal transactions carried out using a Visa smartcard (we restrict the ATM to Visa smartcards for reasons of simplification); • the information system of the bank (Bank IS) to authorize all transactions carried out by a customer using his or her bank smartcard, but also to access the account balance. Bank customer and Cardholder are, furthermore, mutually exclusive. Another solution, which is a little more developed, consists in considering Bank customer as a specialization of Cardholder. Let’s take the cardholder and list the different ways in which they can use the ATM: Cardholder: one of this ways is that it can withdraw money. The Withdraw money use case has two possible primary actors (but they cannot be simultaneous). Another way to express this notion is to consider the Bank customer actor as a specialization (in the sense of the inheritance relationship) of the more general Cardholder actor. A bank customer is actually a particular card holder who has all the privileges of the latter, as well as others that are specific to him or her as a customer. For all use cases appropriate for the bank customer, you must explicitly bring in Bank IS as a secondary actor. But a problem arises for the shared use case, Withdraw money. Indeed, if the primary actor is a Visa card holder, the Visa AS must be called on (which will then be responsible for contacting the IS of the holder’s bank); whereas the ATM will contact the Bank IS directly if it concerns a bank customer. Another solution would be to distinguish two use cases for the withdrawal of money: Withdraw money using a Visa card and Withdraw money using a bank card. This more precise, yet more cumbersome, modelling is easier for the reader of the diagram to grasp. Furthermore, it clearly tells against the use of generalization between actors, which was mentioned beforehand. Indeed, the distinction between the two use cases is contradictory with the attempt at inheritance of the unique Withdraw money case, which had been viewed more highly, while the secondary actors had not yet been added." One of the use cases identified in the text is "Withdraw money using a Visa Card". I need you to analyze the text and to write an appropriate Use Case description of the use case "Withdraw money using a Visa Card". Please take into account the following information when writing the use case description: • “Use Case Name” is any name that is the simplest possible but descriptive too. • "Scenario": Each use case contains the internal activities set which are unique and it is represented as a scenario. • “Triggering event” is basically an event that is initialized by the primary actor to start use case execution. • “Brief description”: is to describe the whole use case description in a short form. • “Actor” can be a human user, a hardware device, or a software system that interacts with the system for goal accomplishment. • “Related use case” represents the associated use cases. • “Stakeholders” are the persons who are all involved in the successful execution of the system. • “Precondition” is a condition that should be met before the use case starts. • “Postcondition” is a condition that should be met after the use case completed successfully. • “Flow of activities” is the tenth compartment of the use case description. o It consists of two columns about the steps performed by the actor and the response of the system. • “Exception conditions” describes the details of the other activities and the exception conditions.

ChatGPT

**ChatGPT**

Use Case Name: Withdraw Money Using a Visa Card

Scenario: A Visa cardholder initiates a request to withdraw money from the ATM.

Triggering Event: The Visa cardholder inserts their Visa smartcard into the card reader.

Brief Description: The use case involves the Visa cardholder interacting with the ATM to withdraw money securely using their Visa smartcard. The system verifies the cardholder's identity, authorizes the transaction through the Visa Authorization System (VISA AS), and dispenses the requested amount.

Actor:

* Primary Actor: Visa Cardholder
* Secondary Actor: VISA AS

Related Use Case: Withdraw Money Using a Bank Card

Stakeholders:

* Visa Cardholder: Initiates the withdrawal transaction.
* VISA AS: Authorizes the transaction and communicates with the bank's Information System (Bank IS) for approval.

Precondition:

* The Visa cardholder has a valid Visa smartcard.
* The ATM is operational.

Postcondition:

* The requested amount is dispensed to the Visa cardholder.
* Transaction details are recorded.

Flow of Activities:

1. Visa Cardholder inserts the Visa smartcard into the card reader. 1.1 System prompts the cardholder to enter their PIN. 1.2 Visa Cardholder enters the PIN. 1.3 System validates the PIN.
2. System contacts VISA AS for authorization. 2.1 VISA AS verifies the cardholder's authorization.
3. Visa Cardholder specifies the withdrawal amount. 3.1 System validates the requested amount.
4. System dispenses the requested amount. 4.1 Visa Cardholder collects the dispensed cash.

Exception Conditions:

* If the PIN is incorrect, display an error message and allow the cardholder to retry.
* If the VISA AS denies authorization, display a transaction declined message.
* If the requested amount exceeds limits, display an error message.
* If the smartcard is invalid or damaged, prompt the cardholder to contact their bank.

Note: The described use case focuses on the specific scenario of a Visa cardholder withdrawing money. It ensures clarity and avoids ambiguity in the interaction between the Visa cardholder, the ATM system, and the Visa Authorization System.

**Analysis:**

The original provided use case is quite comprehensive and well-structured. It covers the main success scenario, alternative sequences, and error sequences in a clear manner. The flow of events is logically organized, making it easy to follow the steps of the interaction between the Visa cardholder, the ATM, and the VISA authorization system.

One notable aspect is the consideration of time in error sequences, such as E4 and E5, where the ATM takes specific actions after a defined time period. This adds a realistic touch to the scenario, addressing potential issues like the card not being taken back or the banknotes not being collected.

Additionally, the incorporation of alternative sequences (A1, A2, A3) covers various scenarios where deviations from the main success scenario might occur, like entering an incorrect PIN, requesting an amount beyond the daily limit, or declining a receipt. This adds depth to the use case, ensuring that potential issues and user choices are considered.

Moreover, the use of clear and concise language in both the main success scenario and error sequences contributes to the overall readability and understanding of the use case. The detailed postconditions provide a clear indication of the expected state after the completion of the use case.

In terms of potential improvements or differences, one consideration might be to explicitly mention the involvement of the Visa AS in the error sequences (E1, E2, E3, E4, E5) where applicable, as these errors might involve communication or actions with the Visa authorization system.

Overall, the original use case is well-constructed, covering various scenarios comprehensively, and incorporating realistic elements to enhance the understanding of the system's behavior in different situations.

**EXERCISE 5🡪**  (EXERCISE 5 UCD) <https://web.uettaxila.edu.pk/CMS/AUT2010/seSREbs/notes%5CLecture_12%20SRE.pdf>

Name: Validate PIN

Summary: System validates customer PIN

Dependency: None

Actors: ATM Customer

Preconditions: ATM is idle, displaying a Welcome message.

Flow of Events: Basic Path

1. Customer inserts the ATM card into the Card Reader.
2. If the system recognizes the card, it reads the card number.
3. System prompts the customer for a PIN number.
4. Customer enters PIN.
5. System checks the expiration date and whether the card is lost or stolen.
6. If the card is valid, the system then checks whether the user-entered PIN matches the card PIN maintained by the system.
7. If PIN numbers match, the system checks what accounts are accessible with the ATM card.
8. System displays customer accounts and prompts the customer for transaction type: Withdrawal, Query, or Transfer.

Validate PIN Use Case -2 Validate PIN Use Case -3

Alternatives:

* If the system does not recognize the card, the card is ejected.
* If the system determines that the card date has expired, the card is confiscated.
* If the system determines that the card has been reported lost or stolen, the card is confiscated.
* If the customer-entered PIN does not match the PIN number for this card, the system re-prompts for PIN.
* If the customer enters the incorrect PIN three times, the system confiscates the card.
* If the customer enters Cancel, the system cancels the transaction and ejects the card.

Postcondition: Customer PIN has been validated.

**PROMPT:** The following description contains many use case: " A bank has several automated teller machines (ATMs), which are geographically distributed and connected via a wide area network to a central server. Each ATM machine has a card reader, cash dispenser, a keyboard/display, and a receipt printer. By using the ATM machine, a customer can withdraw cash from either checking or savings account, query the balance of an account, or transfer funds from one account to another. A transaction is initiated when a customer inserts an ATM card into the card reader. Encoded on the magnetic strip on the back of the ATM card are the card number, the start date, and the expiration date. Assuming the card is recognized, the system validates the ATM card to determine: that the expiration date has not passed, that the user-entered PIN matches the PIN maintained by the system, that the card is not lost or stolen. The customer is allowed three attempts to enter the correct PIN; the card is confiscated if the third attempt fails. Cards that have been reported lost or stolen are also confiscated. If the PIN is validated satisfactorily, the customer is prompted for a withdrawal, query, or transfer transaction. Before a withdrawal transaction can be approved, the system determines that sufficient funds exist in the requested account, that the maximum daily limit will not be exceeded, and that there are sufficient funds available at the local cash dispenser. If the transaction is approved, the requested amount of cash is dispensed, a receipt is printed containing information about the transaction, and the card is ejected. Before a transfer transaction can be approved, the system determines: that the customer has at least two accounts and that there are sufficient funds in the account to be debited. For approved query and transfer requests, a receipt is printed, and the card ejected. A customer may cancel a transaction at any time; the transaction is terminated, and the card is ejected. Customer records, account records, and debit card records are all maintained at the server. An ATM operator may start up and close down the ATM to replenish the ATM cash dispenser and for routine maintenance. It is assumed that functionality to open and close accounts and to create, update, and delete customer and debit card records is provided by an existing system and is not part of this problem. The customer is able to: Withdraws funds from the checking or savings account, Query the balance of an account, Transfer funds from one account to another. Consider that the action of PIN validation is common to all these 3 actions. The ATM customer interacts with the system via the ATM card reader, keyboard/display, cash dispenser, and receipt printer. " One of the use cases identified in the text is "Validate PIN". I need you to analyze the text and to write an appropriate Use Case description of the use case "Validate PIN". Please take into account the following information when writing the use case description: • “Use Case Name” is any name that is the simplest possible but descriptive too. • "Scenario": Each use case contains the internal activities set which are unique and it is represented as a scenario . • “Triggering event” is basically an event that is initialized by the primary actor to start use case execution. • “Brief description”: is to describe the whole use case description in a short form. • “Actor” can be a human user, a hardware device, or a software system that interacts with the system for goal accomplishment. • “Related use case” represents the associated use cases. • “Stakeholders” are the persons who are all involved in the successful execution of the system. • “Precondition” is a condition that should be met before the use case starts. • “Postcondition” is a condition that should be met after the use case completed successfully. • “Flow of activities” is the tenth compartment of the use case description. o It consists of two columns about the steps performed by the actor and the response of the system. • “Exception conditions” describes the details of the other activities and the exception conditions.

ChatGPT

**ChatGPT**

**Use Case Name:** Validate PIN

**Scenario:** A customer initiates the PIN validation process by inserting an ATM card into the card reader.

**Triggering Event:** Customer inserts the ATM card into the card reader.

**Brief Description:** The use case involves validating the customer's Personal Identification Number (PIN) when initiating a transaction. The system checks the validity of the ATM card, ensuring that the expiration date has not passed, the entered PIN matches the system's records, and the card is not reported as lost or stolen. The customer is given three attempts to enter the correct PIN; failure on the third attempt results in the confiscation of the card.

**Actor:** ATM Customer

**Related Use Cases:**

* Withdraw Funds
* Query Balance
* Transfer Funds

**Stakeholders:**

* ATM Customer
* Bank (System)

**Precondition:**

* The ATM card is inserted into the card reader.
* The card is recognized by the system.

**Postcondition:**

* Successful validation allows the customer to proceed with the desired transaction.
* If unsuccessful after three attempts, the card is confiscated.

**Flow of Activities:**

| **Actor** | **System** |
| --- | --- |
| 1. Insert ATM card |  |
|  | 2. Read card information |
|  | 3. Validate expiration date |
|  | 4. Validate PIN (3 attempts allowed) |
|  | 5. If PIN validation successful, proceed |
|  | 6. If PIN validation fails after 3 attempts, confiscate card |
|  | 7. If card reported lost or stolen, confiscate card |
| 8. Enter PIN |  |
|  | 9. Verify entered PIN against system records |
| 10. Verify PIN |  |
| 11. Proceed with transaction or confiscate card if unsuccessful |  |
|  | 12. Eject card (after successful transaction or confiscation) |

**Exception Conditions:**

* If the entered PIN is incorrect, inform the customer and allow additional attempts.
* If the card is reported lost or stolen, confiscate the card.
* If the expiration date has passed, inform the customer and do not proceed with the transaction.
* If the card is not recognized, inform the customer.

**Analysis:**

This use case ensures the secure and accurate validation of the customer's PIN, a crucial step in the ATM transaction process. The flow of activities outlines the interactions between the actor (ATM Customer) and the system, including potential exception conditions that may arise during the validation process.

The original solution is straightforward and efficiently describes the PIN validation process for an ATM transaction. It outlines the essential steps, including card recognition, PIN entry, and potential alternative scenarios like card expiration or being reported lost or stolen. The structure is clear and concise.

In ChatGPT´s response, it was expanded on certain aspects, such as introducing a table for the flow of activities to enhance visual clarity. It also included additional details like a triggering event, related use cases, stakeholders, and a more explicit breakdown of the flow of activities.

While both solutions effectively convey the core concept of validating the PIN for an ATM transaction, ChatGPT´s response aimed to provide a bit more detail and structure, primarily through the use of a table and additional sections for related information. The differences are largely in the presentation and level of detail rather than a substantial deviation from the core content of the original solution.

**EXERCISE 6**🡪 (EXERCISE 12 UCD) <https://web.uettaxila.edu.pk/CMS/AUT2010/seSREbs/notes%5CLecture_12%20SRE.pdf>

Name: Withdraw Funds

Summary: Customer withdraws a specific amount of funds from a valid bank account

Dependency: Include Validate PIN abstract use case

Actors: ATM Customer

Preconditions: ATM is idle, displaying a Welcome message.

Flow of Events: Basic Path

1. Include Validate PIN abstract use case.
2. Customer selects Withdrawal, enters the amount, and selects the account number.
3. System checks whether the customer has enough funds in the account and whether the daily limit will not be exceeded.
4. If all checks are successful, the system authorizes dispensing of cash.
5. System dispenses the cash amount.
6. System prints a receipt showing transaction number, transaction type, amount withdrawn, and account balance.
7. System ejects card.
8. System displays Welcome balance.

Alternatives:

* If the system determines that the account number is invalid, it displays an error message and ejects the card.
* If the system determines that there are insufficient funds in the customer’s account, it displays an apology and ejects the card.
* If the system determines that the maximum allowable daily withdrawal amount has been exceeded, it displays an apology and ejects the card.
* If the ATM is out of funds, the system displays an apology, ejects the card, and shuts down the ATM.

Post-condition: Customer funds have been withdrawn.

**PROMPT:** The following description contains many use case: " A bank has several automated teller machines (ATMs), which are geographically distributed and connected via a wide area network to a central server. Each ATM machine has a card reader, cash dispenser, a keyboard/display, and a receipt printer. By using the ATM machine, a customer can withdraw cash from either checking or savings account, query the balance of an account, or transfer funds from one account to another. A transaction is initiated when a customer inserts an ATM card into the card reader. Encoded on the magnetic strip on the back of the ATM card are the card number, the start date, and the expiration date. Assuming the card is recognized, the system validates the ATM card to determine: that the expiration date has not passed, that the user-entered PIN matches the PIN maintained by the system, that the card is not lost or stolen. The customer is allowed three attempts to enter the correct PIN; the card is confiscated if the third attempt fails. Cards that have been reported lost or stolen are also confiscated. If the PIN is validated satisfactorily, the customer is prompted for a withdrawal, query, or transfer transaction. Before a withdrawal transaction can be approved, the system determines that sufficient funds exist in the requested account, that the maximum daily limit will not be exceeded, and that there are sufficient funds available at the local cash dispenser. If the transaction is approved, the requested amount of cash is dispensed, a receipt is printed containing information about the transaction, and the card is ejected. Before a transfer transaction can be approved, the system determines: that the customer has at least two accounts and that there are sufficient funds in the account to be debited. For approved query and transfer requests, a receipt is printed, and the card ejected. A customer may cancel a transaction at any time; the transaction is terminated, and the card is ejected. Customer records, account records, and debit card records are all maintained at the server. An ATM operator may start up and close down the ATM to replenish the ATM cash dispenser and for routine maintenance. It is assumed that functionality to open and close accounts and to create, update, and delete customer and debit card records is provided by an existing system and is not part of this problem. The customer is able to: Withdraws funds from the checking or savings account, Query the balance of an account, Transfer funds from one account to another. Consider that the action of PIN validation is common to all these 3 actions. The ATM customer interacts with the system via the ATM card reader, keyboard/display, cash dispenser, and receipt printer. " One of the use cases identified in the text is "Withdraw Funds". I need you to analyze the text and to write an appropriate Use Case description of the use case "Withdraw Funds". Please take into account the following information when writing the use case description: • “Use Case Name” is any name that is the simplest possible but descriptive too. • "Scenario": Each use case contains the internal activities set which are unique and it is represented as a scenario . • “Triggering event” is basically an event that is initialized by the primary actor to start use case execution. • “Brief description”: is to describe the whole use case description in a short form. • “Actor” can be a human user, a hardware device, or a software system that interacts with the system for goal accomplishment. • “Related use case” represents the associated use cases. • “Stakeholders” are the persons who are all involved in the successful execution of the system. • “Precondition” is a condition that should be met before the use case starts. • “Postcondition” is a condition that should be met after the use case completed successfully. • “Flow of activities” is the tenth compartment of the use case description. o It consists of two columns about the steps performed by the actor and the response of the system. • “Exception conditions” describes the details of the other activities and the exception conditions.

ChatGPT

**ChatGPT**

Use Case Name: Withdraw Funds

Scenario: A customer initiates a withdrawal transaction at an ATM to access funds from their checking or savings account.

Triggering Event: The customer inserts their ATM card into the card reader.

Brief Description: The customer requests to withdraw funds from their account using the ATM.

Actor: ATM Customer

Related Use Cases: Query Balance, Transfer Funds

Stakeholders: Customer, Bank Server, ATM Operator

Precondition: The ATM is idle, displaying a welcome message, and the customer has inserted a valid ATM card.

Postcondition: The requested amount is dispensed, and a receipt is provided; the ATM returns the customer's card.

Flow of Activities:

| **Actor** | **System** |
| --- | --- |
| 1. Customer inserts ATM card into the card reader. |  |
|  | 2. System reads the card number, start date, and expiration date. |
|  | 3. System validates the card: expiration date, PIN, status. |
| 4. Customer enters the PIN. |  |
|  | 5. System validates the entered PIN. |
|  | 6. System prompts for transaction type: Withdraw, Query, Transfer. |
| 7. Customer selects "Withdraw Funds." |  |
|  | 8. System checks account balance, daily limit, and cash availability. |
|  | 9. If conditions are met, the system approves the withdrawal. |
| 10. Customer enters the desired withdrawal amount. |  |
|  | 11. System validates the amount against daily limits. |
|  | 12. If approved, dispenses the cash. |
|  | 13. Prints a receipt with transaction details. |
|  | 14. Ejects the customer's card. |
| 15. Customer takes the cash and receipt. |  |

Exception Conditions:

* If the system fails to validate the card, eject the card.
* If the entered PIN is incorrect three times, confiscate the card.
* If the requested amount exceeds daily limits, prompt the customer to enter a different amount.
* If the ATM runs out of cash during the transaction, terminate the transaction and eject the card.

This use case ensures that the customer can successfully withdraw funds from their account while considering various scenarios and exceptional conditions.

**Analysis:**

The first solution offered is organized and clearly illustrates the options, postconditions, and sequence of events. The "Validate PIN" abstract use case's explicit dependency inclusion, which highlights the significance of PIN validation prior to the withdrawal process, is one important change. This guarantees a safe transaction and improves the overall logic of the system's clarity. The addition of alternatives, such as managing invalid account numbers or insufficient cash, strengthens the system and completely covers a range of eventualities. The postcondition clearly explains the intended outcome by briefly stating that the withdrawal was completed successfully. Although the two solutions are comparable, the original solution is a little more thorough and explicit representation of the "Withdraw Funds" use case due to its explicit inclusion of the "Validate PIN" requirement and its careful handling of alternate circumstances.

***EXERCISE 7🡪*** <https://www.chegg.com/homework-help/systems-analysis-and-design-in-a-changing-world-6th-edition-chapter-5-problem-2pe-solution-9781133713371>

<https://mygust.com/uploads/BOOK-Systems_analysis_and_design_in_a_changin.pdf>

| **Use Case Name:** | Add a new vehicle to an existing policy. |
| --- | --- |
| **Scenario:** | Add a new vehicle to an existing policy through telephone. |
| **Triggering event:** | Customer wants to buyvehicle. |
| **Brief description:** | Customer gives the car details. If the details are valid, the clerk asked the customer about the coverage and the amount. It is validated by the system. Customer identifies the drivers with the driving time. The policy is updated by the system and it is mailed to the owner with the calculated premium amount. |
| **Actors:** | Clerk  Customer  System |
| **Related use case:** | Add new driver. |
| **Stakeholders:** | Customer  Department of customer service |
| **Preconditions:** | Customer details must be valid.  Existence of “StandardVehicle” Table.  Existence of “StandardCoverage” Table. |
| **Postconditions:** | Object for new vehicle must be created.  The created object should be connected with policy, “StandardVehicle”.  Object for new coverage must be created.  The created object should be connected with vehicle, “StandardCoverage”.  When adding a new driver, the object for new driver must be created.  Driver and the driving time must be updated.  Policy and the premium amount must be updated. |
| **Flow of activities:** | **Actor** | **System** |
|  | 1. Policy number is entered by the clerk.  2. Car details are entered by the clerk.  3. Coverage and amount are entered by the clerk.  4. New driver use case is invoked by the clerk if required.  5. Clerk enters the driver name and driving time.  6. End of process is indicated by the clerk. | 1.1 Policy number is validated by the system.  1.2 System prompts car details.  2.1 Car details are validated by the system.  2.2 System prompts coverage.  3.1 System adds the coverage.  3.2 System validates the amount with the policy limit and updatesit.  3.3 System validates and updates all coverage.  5.1 System adds the driver.  6.1 System computes premium amount.  6.2 System updates the policy.  6.3System mails the policy statement to the customer. |
| **Exception conditions:** | 2.1 Get car details for invalid car details.  3.1 If it is not current policy then request payment and also get other required details.  3.2 If the entered amount is not in limit then request some other amount.  3.3 If the entered coverage details are not valid, get the coverage details again. |  |

The following description includes many use cases: "A customer calls a clerk at the insurance company and gives his policy number. The clerk enters this information, and the system displays the basic insurance policy. The clerk then checks the information to make sure the premiums are current and the policy is in force. The customer gives the make, model, year, and vehicle identification number (VIN) of the car to be added. The clerk enters this information, and the system ensures that the given data is valid. Next, the customer selects the types of coverage desired and the amount of each. The clerk enters the information, and the system records it and validates the requested amount against the policy limits. After all the coverages have been entered, the system ensures the total coverage against all other ranges, including other cars on the policy. Finally, the customer must identify all the drivers and the percentage of time they drive the car. If a new driver is to be added, then another use case—Add new driver—is invoked. At the end of the process, the system updates the policy, calculates a new premium amount, and prints the updated policy statement to be mailed to the policy owner." One of the use cases identified in the text is "Add a new vehicle to an existing policy". I need you to analyze the text and to write an appropriate Use Case description of the use case "Add a new vehicle to an existing policy". Please take into account the following information when writing the use case description: • “Use Case Name” is any name that is the simplest possible but descriptive too. • "Scenario": Each use case contains the internal activities set which are unique and it is represented as a scenario . • “Triggering event” is basically an event that is initialized by the primary actor to start use case execution. • “Brief description”: is to describe the whole use case description in a short form. • “Actor” can be a human user, a hardware device, or a software system that interacts with the system for goal accomplishment. • “Related use case” represents the associated use cases. • “Stakeholders” are the persons who are all involved in the successful execution of the system. • “Precondition” is a condition that should be met before the use case starts. • “Postcondition” is a condition that should be met after the use case completed successfully. • “Flow of activities” is the tenth compartment of the use case description. o It consists of two columns about the steps performed by the actor and the response of the system. • “Exception conditions” describes the details of the other activities and the exception conditions.

ChatGPT

**ChatGPT**

**Use Case Name:** Add a New Vehicle to an Existing Policy

**Scenario:** Adding a new vehicle to an existing insurance policy.

**Triggering Event:** The customer provides information about a new vehicle to be added to their existing insurance policy.

**Brief Description:** This use case involves the process of adding a new vehicle to an existing insurance policy. The customer provides details about the new vehicle, and the system ensures the validity of the entered data. The clerk selects the desired coverage types and amounts, and the system validates the requested coverage against policy limits. The system also checks the overall coverage against all other ranges, considering other vehicles on the policy. The customer must identify all drivers and their driving percentages. If a new driver is involved, the "Add New Driver" use case is invoked. The system updates the policy, calculates a new premium, and generates an updated policy statement for mailing to the policy owner.

**Actor:** Clerk

**Related Use Case:** Add New Driver

**Stakeholders:**

1. Clerk: Initiates the process of adding a new vehicle to the policy.
2. Customer: Provides details about the new vehicle.

**Precondition:** The customer's existing insurance policy is in force and accessible in the system.

**Postcondition:** The system updates the policy with the information about the new vehicle, calculates a new premium, and generates an updated policy statement.

**Flow of Activities:**

| **Actor's Steps** | **System's Response** |
| --- | --- |
| 1. Clerk enters the policy number. | 2. System displays the basic insurance policy for verification. |
| 3. Clerk checks the policy status and premiums. | 4. System validates the current policy status and premiums. |
| 5. Customer provides new vehicle details (make, model, year, VIN). | 6. System ensures the validity of the entered vehicle data. |
| 7. Clerk selects coverage types and amounts for the new vehicle. | 8. System records and validates the coverage against policy limits. |
| 9. System checks overall coverage against all policy ranges. | 10. System ensures total coverage considering other vehicles on the policy. |
| 11. Customer identifies all drivers and their driving percentages. | 12. If a new driver is involved, invoke the "Add New Driver" use case. |
| 13. System updates the policy with new vehicle information. | 14. System calculates a new premium for the updated policy. |
| 15. System prints the updated policy statement. | 16. Clerk mails the updated policy statement to the policy owner. |

**Exception Conditions:**

* If the provided policy number is invalid or not found, the system displays an error message.
* If the system cannot validate the current policy status and premiums, it prompts the clerk to verify the information.
* If the entered vehicle details are invalid or incomplete, the system alerts the clerk to correct the data.
* In case the requested coverage exceeds policy limits, the system notifies the clerk to adjust the coverage amounts.
* If there are issues with updating the policy or calculating the new premium, the system displays an error message, and the clerk contacts support for resolution.

**Analysis:**

The provided solution for the "Add a New Vehicle to an Existing Policy" use case is well-organized and succinct, covering the essential steps involved in the process. One notable aspect is the inclusion of a related use case, "Add new driver," which ensures a modular and scalable approach to handling new drivers within the existing policy. The explicit inclusion of preconditions and postconditions adds clarity to the context and expected outcomes, ensuring a thorough understanding of the system state before and after the use case execution. However, one potential improvement could be to enhance the flow of activities by indicating a clearer transition from the clerk's actions to the system's responses. The solution generated by ChatGPT introduced a slightly more granular breakdown of activities, explicitly stating the actor's steps and the corresponding system responses in separate columns. This approach provides a detailed and structured view of the interaction between the actor and the system. Both solutions effectively address the use case, with the provided solution offering a more streamlined and concise presentation, while the solution generated by ChatGPT provides a more detailed breakdown of actor-system interactions.

***EXERCISE 8:***🡪<https://www.chegg.com/homework-help/systems-analysis-and-design-in-a-changing-world-6th-edition-chapter-5-problem-2tsb-solution-9781133713371> <https://mygust.com/uploads/BOOK-Systems_analysis_and_design_in_a_changin.pdf>

| **Use Case Name:** | Add new resort |
| --- | --- |
| **Scenario:** | Adding a new resort by the clerk |
| **Triggering event:** | Request by a new resort to join with the spring breaks |
| **Brief description:** | A new resort is added with the spring breaks. Accommodation and other facilities are included to the spring breaks. |
| **Actors:** | Clerk  Employees of resort |
| **Related use case:** |  |
| **Stakeholders:** | Management of the spring breaks  Management of the resorts |
| **Preconditions:** | The resort should not exit already |
| **Postconditions:** | Creating a resort  Creating the facilities  Creating the facilities |
| **Flow of activities:** | **Actor** | **System** |
|  | 1. Searching the resort  2.Resort details are entered by the clerk  3.Resort facilities details are entered by the clerk  4.Resort accommodation details are entered by the clerk | 1.1 System shows the resort details  2.1 System adds the resort details  3.1 System adds the facility details  4.1System adds the accommodation details |
| **Exception conditions:** | 1.1 Existence of a new resort |  |

**PROMPT:** The following description includes many use cases: "Spring Breaks 'R' Us (SBRU) is an online travel service that books spring break trips to resorts for college students. Students have booked spring break trips for decades, but changes in technology have transformed the travel business in recent years. SBRU moved away from having campus reps with posted fliers and moved to the web early on.The basic idea is to get a group of students to book a room at a resort for one of the traditional spring break weeks. SBRU contracts with dozens of resorts in key spring break destinations in Florida, Texas, the Caribbean, and Mexico. Its website shows information on each resort and includes prices, available rooms, and special features. Students can research and book a room, enter contract information, and pay deposits and final payments through the system. SBRU provides updated booking information, resort information updates, and travel information for booked students when they log into the site. The resorts also need access to information from SBRU. They need to know about their bookings for each week, the room types that are booked, and so forth. Before the spring break booking season starts, they need to enter information on their resorts, including prices and special features. Resorts need to be paid by SBRU for the bookings, and they need to be able to report and collect for damages caused by spring breakers during their stay. SBRU has recently decided to upgrade its system to provide social networking features for students. It is currently researching possibilities and collecting information from prospective customers about desirable features and functions. From a business standpoint, the idea is to increase bookings by enhancing the experience before, during, and after the trip. Consider the following description of the Booking subsystem. A few weeks before Thanksgiving break, it is time to open the system to new bookings. Students usually want to browse through the resorts and do some planning. After that, when a student or group of students wants to book a trip, the system allows it. Sometimes, a student needs to be added or dropped from the group or a group changes size and needs a different type of room. One month before the actual trip, it is time for the system to send out final payment requirement notices. Students cancel the booking or they pay their final bills. Students often want to look up their booking status and check on resort details. When they arrive at the resort, they need to check in, and when they leave, they need to check out." One of the use cases identified in the text is "Add new resort". I need you to analyze the text and to write an appropriate Use Case description of the use case "Add new resort". Please take into account the following information when writing the use case description: • “Use Case Name” is any name that is the simplest possible but descriptive too. • "Scenario": Each use case contains the internal activities set which are unique and it is represented as a scenario . • “Triggering event” is basically an event that is initialized by the primary actor to start use case execution. • “Brief description”: is to describe the whole use case description in a short form. • “Actor” can be a human user, a hardware device, or a software system that interacts with the system for goal accomplishment. • “Related use case” represents the associated use cases. • “Stakeholders” are the persons who are all involved in the successful execution of the system. • “Precondition” is a condition that should be met before the use case starts. • “Postcondition” is a condition that should be met after the use case completed successfully. • “Flow of activities” is the tenth compartment of the use case description. o It consists of two columns about the steps performed by the actor and the response of the system. • “Exception conditions” describes the details of the other activities and the exception conditions.

ChatGPT

**ChatGPT**

Use Case Name: Add New Resort

Scenario: Adding a new resort to the Spring Breaks 'R' Us (SBRU) system.

Triggering Event: The resort management decides to include a new resort in the SBRU offerings.

Brief Description: The resort management initiates the process of adding a new resort to the SBRU system, providing necessary details such as pricing, special features, and other relevant information.

Actor: Resort Management

Related Use Case: Update Resort Information

Stakeholders:

* Resort Management
* SBRU System Administrators

Precondition:

* The resort must not already be listed in the SBRU system.

Postcondition:

* The new resort is successfully added to the SBRU system.

Flow of Activities:

Actor (Resort Management) | System

1. Resort Management initiates the "Add New Resort" process.
2. System prompts for essential resort details, including pricing and special features.
3. Resort Management enters the required information.
4. System validates the entered information for completeness and accuracy.
5. If the validation is successful, the system adds the new resort to the system.
6. System confirms the successful addition of the new resort.

Exception Conditions: 4. If the entered information is incomplete or inaccurate, the system prompts Resort Management to provide the correct details.

Related Use Case: Update Resort Information (if the resort details need to be modified after the initial addition).

**Analysis:**

The provided solution for the "Add New Resort" use case is straightforward and adequately captures the essential steps involved in adding a new resort to the Spring Breaks 'R' Us (SBRU) system. It highlights the role of the clerk in the process, along with the necessary details such as resort information, facilities, and accommodations. The inclusion of preconditions and postconditions adds a level of clarity to the overall context. However, there are a few areas where the original solution could be enhanced. Firstly, the term "Clerk" is mentioned as the actor, but it might be beneficial to specify whether this refers to a specific role within the resort management or SBRU system administration. Additionally, the flow of activities could be more detailed, breaking down the steps performed by both the actor and the system in a more granular manner. The proposed solution by ChatGPT introduced the concept of related use cases and stakeholders, providing a broader understanding of the system's interactions. It also explicitly mentioned validation steps for the entered information. Both solutions effectively address the use case, with the provided solution being concise and the generated solution providing a slightly more detailed breakdown of activities and interactions.

***EXERCISE 9🡪*** <https://www.usability.gov/how-to-and-tools/methods/use-cases.html#:~:text=A%20use%20case%20is%20a,when%20that%20goal%20is%20fulfilled>.

***PROMPT🡪*** The following description is about a use case about laundry. Below, is an example of a use case regarding laundry in which we can see that: -A housekeeper does laundry on a Wednesday -She washes each load. -She dries each load. -She folds certain items. -She irons some items. -She throws away certain items. Business Rules: Laundry can only be done on Wednesdays. All ironed laundry items get hung on hangers Any laundry item that is irrevocably soiled, shrunken, scorched, etc. , gets thrown out. I need you to analyze the text and to write an appropriate Use Case description of the use case "Housekeeper does Laundry". Please take into account the following information when writing the use case description: • “Use Case Name” is any name that is the simplest possible but descriptive too. • "Scenario": Each use case contains the internal activities set which are unique and it is represented as a scenario . • “Triggering event” is basically an event that is initialized by the primary actor to start use case execution. • “Brief description”: is to describe the whole use case description in a short form. • “Actor” can be a human user, a hardware device, or a software system that interacts with the system for goal accomplishment. • “Related use case” represents the associated use cases. • “Stakeholders” are the persons who are all involved in the successful execution of the system. • “Precondition” is a condition that should be met before the use case starts. • “Postcondition” is a condition that should be met after the use case completed successfully. • “Flow of activities” is the tenth compartment of the use case description. o It consists of two columns about the steps performed by the actor and the response of the system. • “Exception conditions” describes the details of the other activities and the exception conditions.

ChatGPT

**ChatGPT**

Use Case Name: Housekeeper does Laundry

Scenario: On Wednesdays, the housekeeper performs a laundry routine, including washing each load, drying each load, folding specific items, ironing some items, and discarding irreparably damaged items.

Triggering Event: Housekeeper initiates the laundry process on a Wednesday.

Brief Description: The housekeeper engages in the laundry process, washing, drying, folding, ironing, and discarding items according to the specified business rules.

Actor: Housekeeper

Related Use Case: None

Stakeholders: Housekeeper

Precondition: Laundry day is Wednesday.

Postcondition: Laundry is completed, and clothes are appropriately processed. All ironed items are hung on hangers, and irreparably damaged items are thrown out.

Flow of Activities:

| **Actor** | **System** |
| --- | --- |
| Housekeeper starts laundry process. |  |
| Housekeeper loads the washing machine. | System washes the clothes. |
| Housekeeper transfers clothes to the dryer. | System dries the clothes. |
| Housekeeper folds certain items. | System records folded items. |
| Housekeeper irons specific items. | System records ironed items and hangs them on hangers. |
| Housekeeper discards irreparably damaged items. | System updates discarded items. |
| Housekeeper completes the laundry process. |  |

Exception Conditions: If the washing machine or dryer malfunctions, the housekeeper contacts maintenance. If a specific item cannot be folded or ironed, the system notes the exception for manual handling. If an error occurs during the laundry process, the housekeeper may need to repeat certain steps.

**Analysis:**

There are matches between the current solution and the original solution with regard to the fundamental tasks associated with doing laundry, including sorting, washing, drying, folding, and ironing. Both approaches preserve a logical progression, guaranteeing an accurate depiction of the housekeeper's activities on laundry day. ChatGPT´s method, on the other hand, groups related tasks under a generic flow of events and arranges alternate flows for particular cases, providing a more succinct and ordered presentation. ChatGPT´s version also addressed the importance of using a stakeholder named "Housekeeper" in order to better conform to accepted use case modeling practices. A complete understanding can be gained from the original solution's detailed analysis of alternate flows and exceptions, but a high-level use case description may not require such granularity. The new provided approach aimed for clarity while keeping essential information for clear understanding and communication.

***EXERCISE 10🡪*** <https://www.chegg.com/homework-help/systems-analysis-and-design-in-a-changing-world-6th-edition-chapter-5-problem-1pe-solution-9781133713371> <https://mygust.com/uploads/BOOK-Systems_analysis_and_design_in_a_changin.pdf>

| **Use Case Name:** | Supply items to contractor |
| --- | --- |
| **Scenario:** | Supply items to in-store contractor |
| **Triggering event:** | Existing contractor wants to buy items. |
| **Brief description:** | A contractor buys items. The clerk at the checkout desk opens a new sale ticket and then adds the item details into the contractor’s account and verifies the credit limit to add the amount of purchase. If the contractor wants the ticket details are printed. |
| **Actors:** | Clerk |
| **Related use case:** | Might be called by the shopping through online use case,  Card verification use case. |
| **Stakeholders:** | Clerk  Department of accounts  Department of sales |
| **Preconditions:** | Contractor account must be existing.  Credit limit must be valid.  Price of each inventory items must be existing. |
| **Postconditions:** | New sale must be created.  Contractor account must be updated.  Number of inventory items must be updated. |
| **Flow of activities:** | **Actor** | **System** |
|  | 1. Contractor name and ID is entered by the clerk.  2. Clerk opens a new sale ticket and adds the inventory item.  3. End of sale is indicated by the clerk.  4. If contractor needs receipt then the clerk requests the print of receipt. | 1.1 The contractor account is validated by the system.  1.2 The contractor details with credit balance are displayed by the system.  1.3 System prompts to open sale ticket.  2.1 System adds the price of item with the total.  2.2 System automatically updates the inventory items.  3.1 System computes final sum.  3.2 System subtracts the final sum from the credit balance to update the account of contractor.  4.1 The receipt is printed by the system. |
| **Exception conditions:** | 1.1 If the credit balance of the contractor is below limit then allow payment through cash or card.  2.1 If the price for an inventory item is missing, get the amount and enter it manually. |  |

**PROMPT:** The following description contains many use case: "Quality Building Supply Sales Process: Contractor Sales: The contractor brings materials to the checkout desk designated for contractors. The clerk enters the contractor’s name into the system. The system displays the contractor’s information, including their current credit standing. The clerk opens a new ticket (sale) for the contractor in the system. Each item to be purchased is scanned by the clerk. The system finds the price of each item and adds it to the ticket. At the end of the purchase, the clerk indicates the completion of the sale. The system compares the total amount against the contractor’s current credit limit. If the total amount is acceptable, the system finalizes the sale. An electronic ticket for the items is created, and the contractor’s credit limit is reduced by the amount of the sale. Some contractors prefer to keep a record of their purchases, so they may request that ticket details be printed. Others may not be interested in a printout. General Public Sales: The sale to the general public is simply entered into the cash register. A paper ticket is printed as the items are identified. Payment can be made by cash, check, or credit card. The clerk must enter the type of payment to ensure that the cash register balances at the end of the shift. For credit card payments, the system prints a credit card voucher that the customer must sign. This process ensures efficient and tailored service for both contractors and the general public, maintaining accurate records and providing flexibility based on customer preferences. " One of the use cases identified in the text is "supply items to contractor". I need you to analyze the text and to write an appropriate Use Case description of the use case "supply items to contractor". Please take into account the following information when writing the use case description: • “Use Case Name” is any name that is the simplest possible but descriptive too. • "Scenario": Each use case contains the internal activities set which are unique and it is represented as a scenario . • “Triggering event” is basically an event that is initialized by the primary actor to start use case execution. • “Brief description”: is to describe the whole use case description in a short form. • “Actor” can be a human user, a hardware device, or a software system that interacts with the system for goal accomplishment. • “Related use case” represents the associated use cases. • “Stakeholders” are the persons who are all involved in the successful execution of the system. • “Precondition” is a condition that should be met before the use case starts. • “Postcondition” is a condition that should be met after the use case completed successfully. • “Flow of activities” is the tenth compartment of the use case description. o It consists of two columns about the steps performed by the actor and the response of the system. • “Exception conditions” describes the details of the other activities and the exception conditions.

ChatGPT

**ChatGPT**

**Use Case Name:** Supply Items to Contractor

**Scenario:** The contractor arrives at the designated checkout desk for contractors, initiating the process of purchasing construction materials.

**Triggering Event:** Contractor brings materials to the checkout desk.

**Brief Description:** This use case involves the clerk facilitating the purchase of construction materials by a contractor. The system records the items, calculates the total cost, and checks the contractor's credit standing before finalizing the sale.

**Actor:** Clerk, System, Contractor

**Related Use Case:** None

**Stakeholders:** Clerk, Contractor

**Precondition:**

* The contractor is at the designated checkout desk.
* The clerk has access to the system.

**Postcondition:**

* The sale is successfully completed.
* Items are supplied to the contractor.
* Contractor's credit limit is updated.

**Flow of Activities:**

Actor (Clerk):

1. Enters the contractor’s name into the system.

2. Displays contractor information, including credit standing.

3. Opens a new sale ticket in the system.

4. Scans each item to be purchased.

5. System finds the price of each item and adds it to the ticket.

6. Indicates the completion of the sale.

System:

1. Compares the total amount against the contractor’s credit limit.

2. If acceptable, finalizes the sale.

3. Creates an electronic ticket for the items.

4. Reduces the contractor’s credit limit by the amount of the sale.

Actor (Contractor):

1. May request a printed ticket detailing the purchase.

**Exception Conditions:**

1. If the total amount exceeds the contractor's credit limit, the system alerts the clerk, and further action is taken.
2. If there are issues with the system or scanning process, the clerk may need to resolve technical problems or seek an alternative solution.
3. If the contractor disputes the sale or needs assistance, the clerk follows established procedures for resolution.
4. If the contractor requests a printed ticket, the clerk fulfills the request based on customer preference.

This use case ensures a smooth process for supplying items to contractors, considering their unique needs and maintaining accurate records.

**Analysis:**

With a few significant exceptions, the offered solution has a structure that is identical to the original solution. A deeper understanding of the process is provided by the revised solution, which breaks down the processes followed by the clerk and the system in more precise detail. The updated solution also incorporates particular system interactions, such prompts, automatic updates, and validations, giving a more thorough understanding of the system's function in the use case. By taking into account probable unusual events, the provision of alternative scenarios, which specifically address credit limit concerns and missing item prices, improves the use case's robustness. Overall, the method is still explained in a clear and comprehensive manner in the updated solution, which makes it more detailed and informative.