ITSS SOFTWARE DEVELOPMENT / SOFTWARE DESIGN AND CONSTRUCTION

Assignment 2. Use case specification

Lecturer: NGUYEN Thi Thu Trang, trangntt@soict.hust.edu.vn

1. SUBMISSION GUIDELINES

When you want to submit your individual work for the Case Study, you have to push your work to your individual GitHub repository, complied with the naming convention "TeamName-StudentID.StudentName" (e.g. "ISD. 20232-20202012.HoangNghiaPhu").

For this lab, you have to turn in your work twice before the following deadlines:

- **Right after class**: Push all the work you have done during class time to Github.
- **10 PM the day before the next class**: Create a branch named "***release/lab02***" in your GitHub repository and push the full submission for this lab, including in-class tasks and homework assignments, to this branch.

2. USE CASE SPECIFICATION

In this lab, we continue with the requirement modeling and try it ourselves with Use case diagram, Flow of events, activity diagrams, and use case (UC) specification for the Case Study AIMS. You are asked to work individually for this section, and then put all your file(s) and directories to a parent directory, namely "Requirement Analysis". After that, push your commit to your individual repository before the announced deadline.

In use case models, a use case describes a **flow of events** which is *performed by the software* and *yields an observable result of value to a particular actor*.

In this subsection, we would use the two use cases, UC "Place Order" and UC "Pay Order", to demonstrate how we can make a flow of events or use case specification. Our sample solution is UC "Place Rush Order" is extended from the UC "Place Order". But you can consider and do the following tasks with another solution: UC "Place Rush Order" are in UC "Place Order", and there is no UC "Place Rush Order". **The below guidelines are only samples – not obligation, you have to verify, update as you wish and do by yourself for your submissions.**

3. USE CASE SPECIFICATION FOR "PLACE ORDER"

In the AIMS Project, UC "Place Order" describes the interaction between customers and AIMS software when the customer wishes to place order.

3.1. FLOW OF EVENTS

Naturally, we describe how the use case starts and ends to gain the purpose of a use case, and we may think of a basic flow of the events for UC "Place Order" as follows.

Step 1. Customer requests to place order in the cart
Step 2. AIMS software checks the availability of products in the cart
Step 3. AIMS software displays the form of delivery information with order information
Step 4. Customer enters and submits delivery information (see Table 1)
Step 5. AIMS software calculates and updates order information with shipping fees (see Table 2)
Step 6. The customer asks to pay order
Step 7. The AIMS software calls UC "Pay order"
Step 8. The AIMS software creates and saves a new order
Step 9. The AIMS software makes the cart empty
Step 10. The AIMS software sends email about the order notification and information
Step 11. The AIMS software displays the successful order notification, the order and the transaction information (see

1.	Total			2,346,600

Step 12. Table 3).

We specify what happens when, for each action. Remember this text will be used to identify test cases. However, things hardly ever go as planned. Taking the above flow as our initial basic flow, we would try to analyst and improve our flow of events for UC "Place Order".

Question: Should we trust any input to an action/event?

It depends on the question or the request and the way it has been asked or responded. If the response must comply with a rule or a regulation, we must question the validity of the input. In case of humans, they tend to follow the given

instructions in an incorrect manner even after several trials. In cases of non-human actors, there are a lot of factors that heavily affect the input from other systems, e.g., noise in transmission. Hence, we need to give them the chance to try repeatedly until their input is valid or within an acceptable limitation to be at least. However, there are cases in which we must take the input for granted since it is unnecessary or beyond the control of the system. To illustrate, in the form of delivery information, we should not check if the user has filled the real name of the receiver in the field for receiver name, yet we must not let the user left it blank.

Question: What data is exchanged between actor and use case and between use case and use case?

In this use case, customer needs to provide with delivery information. Some or all attributes of these information may play critical roles in input validation, so we need to specify the attributes of the input. To illustrate, the input data of delivery information may include these data fields:

Table 1- Input data of delivery information

No	Data fields	Description	Mandatory	Valid condition	Example
1.	Receiver Name		Yes		Do Minh Hieu
2.	Phone Number		Yes	10 digits	0987654321
3.	Province	Choose from a list	Yes		Hanoi
4.	Address		Yes		12, 34 Alley of Tran Thai Tong street, Cau Giay district
5.	Shipping instructions		No		

We also need to specify which the output to the actor(s) since it is the main factor that impacts on the input from actor(s). For instance, the output data are shown in the following table.

Table 2-Output data of order information and shipping fee

No	Data fields	Description	Display format	Example
2.	Title	Title of a media product		DVD Phim Vượt ngục
3.	Price	Price of the corresponding media product	▪ Comma for thousands separator ▪ Positive integer ▪ Right alignment	123,000
4.	Quantity	Quantity of the corresponding media	▪ Positive integer ▪ Right alignment	2
5.	Amount	Total money of the corresponding media	▪ Comma for thousands separator ▪ Positive integer ▪ Right alignment	246,000
6.	Subtotal	Total amount of all products in the order		2,316,600
7.	Shipping fee			30,000
8.	Total			2,346,600

Table 3-Output data of general information of order and transaction info

No	Data field	Description	Display format	Example
1.	Customer name			Do Minh Hieu
2.	Phone number			0987654321
3.	Province			Hanoi
4.	Address			12, 34 Alley of Tran Thai Tong street, Cau Giay district
5.	Total amount		Right alignment Vietnamese currency (VNĐ) Vietnamese locale	1.200.000 VNĐ
6.	Transaction ID			
7.	Transaction content			

No	Data field	Description	Display format	Example
8.	Transaction date		dd/mm/yyyy	05/10/2023

Note that we do not describe the details of the user interface unless it is necessary to understand the behavior of the system. Specifying user interface details too early will limit design options.

Now, we can finally validate the data. For list of media in the cart, we need to check if a media is out-of-stock. For delivery information, we need to check if a mandatory field is left blank and valid condition for the phone number. Thus, we need insert at least two more events into the flow so as to validate the two corresponding inputs.

After validation, in case there is an exception, the flow cannot continue normally. Consequently, we need alternative flows or sub-flows for the next events in these cases. For instances, the sub-flows for UC "Place Order" is shown as follows.

The alternative flows of events of the use case "Place order" are illustrated (may be not complete) in Table 3.

Table 4 - Alternative flow of events for UC "Place Order"

No	Location	Condition	Action	Resume location
1.	At Step 3	If the products are not available	▪ The AIMS software notifies that the the products in the cart are not available and stay at the use case "View cart"	Use case ends
2.	At Step 5	If the delivery info is invalid	▪ AIMS software notifies that the delivery info is invalid (blank or wrong format)	At Step 3
3.	At Step 5	If the user chooses to place a rush order	▪ AIMS software inserts use case "Place rush order"	At Step 6
4.	At Step 8	If the order payment is not successul or goes back from payment	▪	At Step 5

The last questions are *what we should save and when we save it*.

By saving the data, we can save a lot of time and efforts for us, the system, and the users. To illustrate, the customer cannot finish placing order for some reasons. Thus, we can save some information for later such as the list of media in cart, so that the customer does not have to add them to the cart again.

3.2. USE CASE SPECIFICATION

Finally, we may provide the pre-condition and the post-condition. For example, the pre-condition for UC "Place Order" can be "There is at least one item in the cart." A post-condition can be "a new order is created, and its information is sent via email to the customer or nothing happens if payment is not successful".

Template of Use Case Specification is shown as below.

Use Case "Name of use case"

1. Use case code

UC00X

2. Brief Description

This use case describes the interaction between <actor(s)> and <name_of_the_system> when <actor(s)> wish(es) to ...

3. Actors

3.1 Name of Actor 1

4. Preconditions

5. Basic Flow of Events

1. Actor ...
2. Software displays ... (see Table T).
3. ...

6. Alternative flows

Table N-Alternative flows of events for UC Place order

No	Location	Condition	Action	Resume location
1.	At Step S	If ...	▪ Action 1	Resumes at Step Q
2.	At Step O	If ...	▪ Action 2	Use case ends

7. Input data

Table A-Input data of …

No	Data fields	Description	Mandatory	Valid condition	Example
1.					

8. Output data

Table B-Output data of …

No	Data fields	Description	Display format	Example
1.				

9. Postconditions

For this part, given the above suggestion, you are asked to make a use case specification for UC "Place Order" by using the template. Remember to validate data and save information if need be. When you finish the task of this part, please export your work to a PDF file, namely "Use case specification – Place Order." Then put both files in the directory "Use case specification".

4. USE CASE SPECIFICATION FOR "PAY ORDER"

In this part, you are asked to fill in the provided template for the use case specification of UC "Pay Order". When you finish the task of this part, please export your work to a PDF file, namely "Use case specification – Pay Order", in the directory "Use Case Specification".

This use case describes the interactions between the AIMS software with the customer and VNPay when the customer desires to pay order. An example for the basic flow of events is listed as follows.

Step 1. AIMS software displays the invoice (see Table 5)
Step 2. Customer asks to pay the invoice

Step 3. AIMS software redirects to VNPay
Step 4. VNPay sends payment result to VNPay
Step 5. AIMS software saves invoice and payment transaction
Step 6. VNPay notifies the payment result

The alternative flows of events of this use case are illustrated in the following table.

No	Location	Condition	Action	Resume location
1.	At Step 5	If the customer cancels the payment transaction	▪	At Step 1

In this use case, the output data when displaying the invoice is shown in the following tables (the rows with green shading are repeated for all media products in the cart/invoice).

Table 5-Output data of invoice

No	Data fields	Description	Display format	Example
1.	Title	Title of a media product		DVD Phim Vượt ngục
2.	Price	Price of the corresponding media product	▪ Comma for thousands separator ▪ Positive integer ▪ Right alignment	123,000
3.	Quantity	Quantity of the corresponding media	▪ Positive integer ▪ Right alignment	2
4.	Amount	Total money of the corresponding media	▪ Comma for thousands separator ▪ Positive integer ▪ Right alignment	246,000
5.	Subtotal Before VAT	Total price of products in the cart before VAT	▪ Comma for thousands separator ▪ Positive integer ▪ Right alignment	2,106,000
6.	Subtotal	Total price of products in the cart with VAT		2,316,600
7.	Shipping fees			30,000

No	Data fields	Description	Display format	Example
8.	Total	Sum of subtotal and shipping fees		2,346,600
9.	Currency			VND
10.	Name			Do Minh Hieu
11.	Phone number			0987654321
12.	Province			Hanoi
13.	Address			12, 34 Alley of Tran Thai Tong street, Cau Giay district
14.	Shipping instructions			

5. USE CASE SPECIFICATION FOR "PLACE RUSH ORDER"

You are asked to update the use case specification for UC "Place order" and/or ***to make another use case specification by using the provided template*** (e.g., you can model the relationship between UC "Place Rush Order" and UC "Place Order" as an extension) for the additional UC "Place Rush Order".

When you finish the task of this part, please export your work to a PDF file, namely "Use case specification – Place Order with Place Rush Order" Then put your work and the exported file in the directory "Use Case Specification".

Note: If you consider "Place Rush Order" in the UC "Place Order", you donot complete this task. However, remember to include flow of events and other information of "Place Rush Order" inside the UC "Place Order".