# 

# **1.** **Task 1: Requirement elicitation (1.1, 1.2)**

## **1.1** **Domain Context (Quốc Thăng)**

Student Smart Printing Service (HCMUT\_SSPS) is a web-based app or a mobile app for providing students, who have HCMUT\_SSO authentication, service to print their documents under the management of Student Printing Service Officer (SPSO) in its campuses. Each printer has an ID, brand/manufacturer name, printer model, short description, location (campus name, building name, room number).

The Student Smart Printing Service (HCMUT\_SSPS) is a web and mobile application designed to provide HCMUT students with a convenient document printing service. The application is intended for internal use by HCMUT students only, and therefore, it should be customized to best meet their specific needs. The service is managed by the Student Printing Service Officer (SPSO), who requires the system to be designed with management functionalities in mind. The service must be available across all HCMUT campuses.

## **1.2** **Stakeholders and Needs (Thanh Lâm + Trung Kiên)**

The stakeholders of the HCMUT Student Printing Service System (SPSS) include a variety of groups with distinct roles and expectations.

End-users primarily consist of students at HCMUT, who rely on SPSS to print documents conveniently. The system must allow them to log in, view their printing history, and access these services easily. Focus on usability and accessibility.

The Student Printing Service Officer (SPSO) acts as the manager of the system. The SPSO must be provided with tools to monitor and manage user activities, printer status, and generate comprehensive reports on system usage.

BKPay system is responsible for processing payments by connecting with OCB bank accounts used by students. The system must ensure secure, reliable transactions while integrating with the HCMUT\_SSPS to offer safe payment options and transactions.

Finally, the HCMUT\_SSO authentication service is vital for user authentication and access control within the SPSS. This service would require secure login processes, efficient management of authentication data, and the protection of user credentials through robust security measures, providing a smooth and secure user experience across the system.

- End-users: Students of HCMUT: The SSPS should

- Provide services for printing documents

- Log in and view history as needed

- Be easy and convenient to use

- Manager: The Student Printing Service Officer (SPSO) should

- Provide features to manage students using the service

- As well as features to manage the printers in the system

- Generate reports of the using of the printing system

-

- Developer: Group 03 - CC02

- A clear specification

- Receive help from SSO and BKPay managers

- BKPay system

- Connect to the OCB bank.

- Students use their OCB’s accounts for paying.

- Secure integration with HCMUT\_SSPS for processing payments.

- Ensure transaction security and traceability.

- HCMUT\_SSO authentication service

- Secure authentication of all users accessing HCMUT\_SSPS.

- Integration with the HCMUT\_SSPS system for access control.

- Manage authentication data securely and efficiently.

- Ensure smooth user login experience.

- Provide robust security measures to protect user credentials.

## **1.3** **Benefits of the System (Thanh Lâm + Trung Kiên)**

The HCMUT Student Smart Printing Service (HCMUT\_SSPS) offers various benefits to its stakeholders. For students, it provides convenience by allowing them to print documents remotely, manage print settings, and monitor their printing usage through logs, while also controlling costs with default print quotas and the option to purchase additional pages via BKPay. The Student Printing Service Officer (SPSO) benefits from centralized management, including the ability to monitor printer usage, control system configurations, and generate automated reports for better resource allocation and oversight. Developers enjoy the system's scalability, ease of integration with existing university services like HCMUT\_SSO and BKPay, and straightforward maintenance. The BKPay system sees increased usage due to frequent student transactions for additional printing pages, while the HCMUT\_SSO authentication service ensures secure access for all users, reinforcing its role in the campus's digital infrastructure. Overall, the system enhances efficiency, security, and accessibility for all stakeholders involved.

## **1.4** **Functional Requirements (Đức Nghĩa)**

- Students:

- The system must allow students to upload document files onto the system.

- The system must allow students to choose a printer from a list.

- The system must allow students to configure the printing properties.

- The system must allow students to print documents.

- The system must allow students to view their personal history

- The system must provide a summary report for students showing the total number of pages they have printed, categorized by paper size

- The system must display the student’s current page balance in real-time

- The system must allow students to buy additional printing pages

- The system must notify students with a warning when their account page balance is low or has run out.

- SPSO:

- The system must allow the SPSO to configure and manage the permitted file types

- The system must allow the SPSO to view the detailed printing history of students and printers for a specified time period

- The system must allow the SPSO to manage printers by adding new printers to the system, and enabling or disabling existing printers.

- The system must allow the SPSO to manage the system configurations.

- The system must generate and allow the SPSO to view monthly and yearly reports

- SSO:

- The system must verify the user's identity before accessing to any system features.

- The system must store the session start and end times for each student using the system

- The system must retrieve user roles as student or SPSO during verifying.

- The system must provide the time session of students to the SPSO for monitoring and administrative purposes.

- The system must notify with a warning when they fail to verify.

- BKPay:

- The system must allow the user (student) to choose the number of page they want to buy

- The system must automatically calculate the total cost based on the number of pages selected

- The system must display available payment methods and a guide that explains how to complete the payment

- The system must receive confirmation of successful payment once the user completes the transaction.

- The system must automatically add the purchased pages to the user’s current page balance when the user completes the transaction.

## **1.5** **Non- Functional Requirements (Gia Bảo)**

Performance requirements

* SSPS System can at least handle the number of concurrent requests equal to the number of available printers.

Security requirements

* Students can only log in as “student authentication role”, and that they are granted limited privileges and can only access specific services.
* Student personal information is protected, and only some of their information is visible to authorized SPSO personnel and the printer currently performing their print order. These information are (1) student name, (2) student ID, (3) information about their current printing order.
* SPSO personnels can only log in as “SPSO authentication role”, and that they are granted limited privileges and can only access specific services.

Availability requirements

* The SSPS System can be accessed by students and can perform operations in common Vietnamese office hours (from 7:30AM to 12AM and from 1:30PM to 5PM).
* The SSPS System can be interfaced with in the form of a web browser application (Chrome, Firefox, and more if possible) and a mobile application (Android, iOS).

Usability requirements

* SSPS applications should support English as the primary human language.
* SSPS applications support light theme as the primary color theme.

Scalability requirements

* SSPS System provides a set of private application programming interface (SSPS API)
* SSPS System can accept new modules that require little change to the core system; these new modules must follow the SSPS API.

Compatibility requirements (inferred none, so this is a contribution)

* SSPS API versions should follow the semantic versioning scheme (SemVer)[[1]](#footnote-0)

Maintainability requirements

* SSPS API can only roll-out a major update after at least one year since the previous update.

2. Task 2. Use-case Diagram

***DEADLINE: Đến hết 28/09/2024 (Thứ 7)***

## **2.1 Use-case diagram for the whole system (Trung Kiên)**

## 

## **2.2** **Use-case Diagram for Modules**

### 2.2.1. Student Printing Module (Trung Kiên)

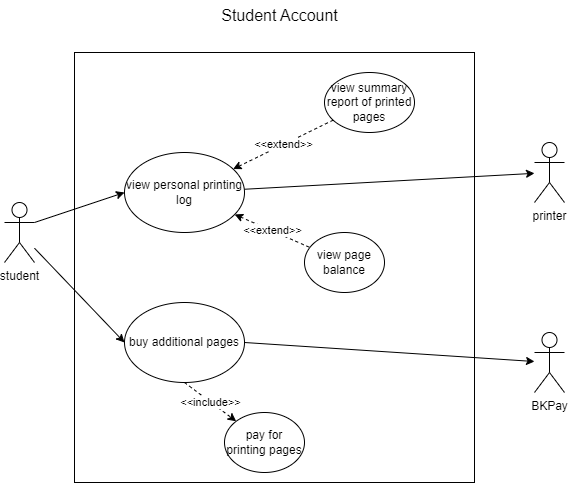
Use cases included:

* Uploading document,
* choose printer from list,
* configure printing properties

### 2.2.2. Student Account Module (Đức Nghĩa)

Use cases included:

* View personal printing history,
* view summary report,
* View page balance
* Purchasing additional pages



### 2.2.3. Printers Managing Module for SPSO (Gia Bảo)

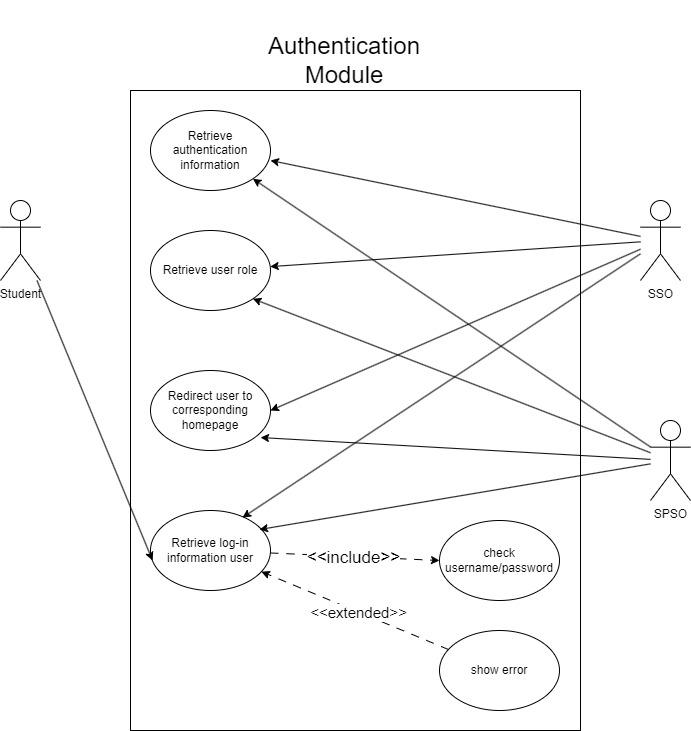
Use cases included:

* Viewing printing history of printers
* Managing printers (adding, enabling, disabling)
* Mange system config
* View monthly/yearly report

### 2.2.4. Authentication Module (Thanh Lâm)

Use cases included:

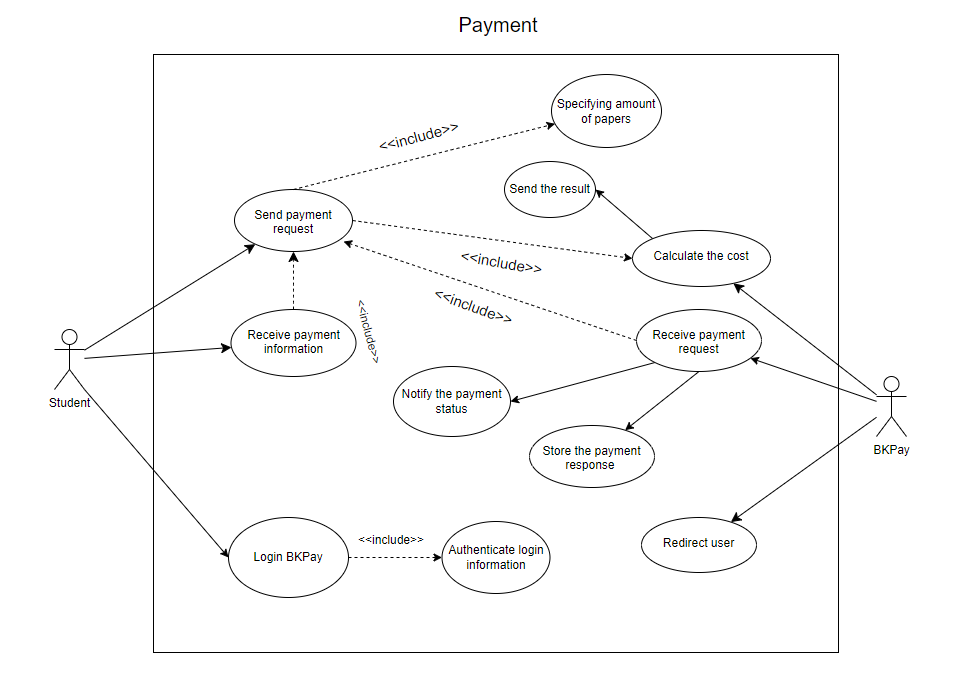
* Retrieve authentication information from SSO
* Receive log in information from user



### 2.2.5. Payment Module (Quốc Thăng)

Use cases included:

* Calculate the cost based on user specified number of pages
* Send payment request (with cost) to BKPay
* Redirect user to BKPay log in page
* Receives payment information from BKPay



## **2.3** **Use-case Description**

### 2.3.1 Use case diagrams for Student Printing Module

| Name: | Uploading document | | |
| --- | --- | --- | --- |
| Created by: | Lê Trung Kiên | Date Created: | 28-07-2024 |
| Primary actor: | Student | Secondary actor: |  |
| Description: | The student choose a file from their system to upload to the system | | |
| Trigger: | Student select “Upload document” | | |
| Preconditions: | PRE-1 User logged in to the system as “Student” | | |
| Postconditions: | POST-1 Student’s file uploaded to the system  POST-2 Operation stored in log | | |
| Normal flow: | **Students upload a file to the system**   1. Student request to upload file to the system 2. A window pop up and let the student choose a file from their system 3. File is uploaded to the system | | |
| Exception flow: |  | | |

| Name: | Print document | | |
| --- | --- | --- | --- |
| Created by: | Lê Trung Kiên | Date Created: | 28-07-2024 |
| Primary actor: | Student | Secondary actor: | Printer |
| Description: | Students request to print a document. The system lets them choose from a list of printers, specifying the printing properties. Then if the users have sufficient page balance. The system send a request to the selected printer to start printing. | | |
| Trigger: | Student requests to print a document. | | |
| Preconditions: | PRE-1 User logged in as “Student”  PRE-2 Student has sufficient page balance  PRE-3 Student has file uploaded  PRE-4 There are printers online | | |
| Postconditions: | POST-1 Printing request sent to selected printer  POST-2 Printing request recorded on log | | |
| Normal flow: | **Student request to print a document**   1. Student select a printer by specifying its ID 2. Student choose one of their uploaded files 3. Student specifying the printing properties 4. System check student page balance 5. System sends a request with printing information to the selected printer 6. System confirm the print is successful 7. The request is recorded on log | | |
| Exception flow: | **Student has insufficient page balance**   1. Student receives a notification that their account has insufficient page balance 2. System shows an option to redirect the user to page purchase. | | |

### 2.3.2

| Name: | Student sends payment request | | |
| --- | --- | --- | --- |
| Created by: | Mai Quốc Thăng | Date created: |  |
| Primary actor: | Student | Secondary actor: | BKPay |
| Description: | The student enters the amount of A4 papers then BKPay system calculates the cost of the given amount of papers and returns it to the student. The student sends the payment request to BKPay. | | |
| Trigger: | Student wants to buy more papers | | |
| Preconditions: | - Student’s identity has been authenticated by HCMUT\_SSO  - Student already logs in the BKPay system  - Student’s balance is sufficient with the cost  - BKPay systems is online | | |
| Postconditions: | - Payment response is sent back to the student immediately | | |
| Normal flow: | 1. Student enters the amount of papers to buy  2. BKPay system returns the amount of money needed to purchase  3. Student confirms if the current balance is sufficient to accept the payment  4. Student sends the payment request to the BKPay system.  5. BKPay system receives the student’s payment request.  6. BKPay system confirms the payment and start the transaction.  7. BKPay system sends the payment response back to the student | | |
| Exception flows: | 1a. Student enters invalid character  - Notifies the student  2a BKPay system fails to send back the amount of money for the purchasing  3a Student cancels the payment  5a BKPay system fails to receive the student’s payment request.  7a BKPay system fails to response about the payment to the student | | |

| Name: | Student logs in the BKPay system | | |
| --- | --- | --- | --- |
| Created by: | Mai Quốc Thăng | Date created: |  |
| Primary actor: | Student | Secondary actor: | BKPay |
| Description: | The student enters the school’s provided email address and the password. The BKPay system then verifies the given information and responds to the student about the login status. | | |
| Trigger: | Student wants to login the BKPay system | | |
| Preconditions: | - Student’s identity has been authenticated by HCMUT\_SSO  - Student’s email and password are authenticated by BKPay system | | |
| Postconditions: | - Student’s login response is sent to the student  - BKPay system starts a new session for the student | | |
| Normal flow: | 1. Student enters the email and password  2. Student send the login request  3. BKPay system receives the login request.  4. BKPay system authenticates the request and store the login response  5. BKPay system sends the login response to the student. | | |
| Exception flows: | 1a. Student’s information is incorrect or non-existent in the BKPay system.  - Notifies the user  3a. BKPay system fails to receive the login request  4a. BKPay system fails to authenticate the login request.  5a. BKPay system fails to send the login response to the student | | |

| Name: | Student receives payment information | | |
| --- | --- | --- | --- |
| Created by: | Mai Quốc Thăng | Date created: |  |
| Primary actor: | Student | Secondary actor: | BKPay system |
| Description: | The student requests to view details about the successful payment request. The system responds to the student with a list of payment with the details for each payment. | | |
| Trigger: | Student wants to view the payment information | | |
| Preconditions: | - Student made at least 1 transaction before  - Student’s identity has been authenticated by HCMUT\_SSO  - Student already logs in the BKPay system | | |
| Postconditions: | - List of payments with its details is sent back to the student immediately | | |
| Normal flow: | 1. Student requests to view the payment information  2. The system accesses and stores the student’s payment history  3. The system sends the payment history to the student | | |
| Exception flows: | 2a. The system fails to access and store the student’s payment history  3a. The system cannot send the payment history to the student | | |

| Name: | BKPay system calculates the cost | | |
| --- | --- | --- | --- |
| Created by: | Mai Quốc Thăng | Date created: |  |
| Primary actor: | BKPay system | Secondary actor: | Student |
| Description: | The BKPay system calculates the total cost of the amount of A4 papers given by the student. BKPay system then displays the result at the payment request | | |
| Trigger: | After student specifies the amount of A4 papers for the payment request | | |
| Preconditions: | - Student enters the amount of papers in the payment request  - BKPay system is online  - Student already logs in the BKPay system | | |
| Postconditions: | - BKPay system displays the calculation result at the payment request immediately | | |
| Normal flow: | 1. Student enters the amount of papers in the payment request  2. BKPay system receives the amount of papers from the payment request.  3. BKPay system calculates the total cost based on the received number.  4. BKPay stores and displays it at the student’s payment request. | | |
| Exception flows: | 1a. Student enters invalid character in the amount field  2a. BKPay system fails to receive the amount of papers from the payment request.  4a. BKPay system fails to display the result at the student’s payment request. | | |

| Name: | BKPay system receives payment request | | |
| --- | --- | --- | --- |
| Created by: | Mai Quốc Thăng | Date created: |  |
| Primary actor: | BKPay system | Secondary actor: | Student |
| Description: | The BKPay system receives the payment request made by the student. It processes the request and performs a transaction. If the transaction succeeds then the BKPay system notifies the student and stores the payment response (history) to the system. | | |
| Trigger: | The student makes a payment request. | | |
| Preconditions: | - Student makes a payment request  - BKPay system is online  - Student already logs in the BKPay system | | |
| Postconditions: | - BKPay system notifies the student about the transaction status  - BKPay system stores the transaction history in the system | | |
| Normal flow: | 1. Student sends a payment request  2. BKPay system receives the payment request from the student  3. BKPay system processes the request and starts the transaction.  4. BKPay system processes the transaction.  5. BKPay system notifies the student about the transaction status.  6. BKPay system stores the transaction history in the system | | |
| Exception flows: | 2a. BKPay system fails to receive the payment request from the student  3a. BKPay system fails to perform a transaction.  5a. BKPay system fails to notify the student about the transaction status.  6a. BKPay system fails to store the transaction history in the system | | |

| Name: | BKPay system redirects student | | |
| --- | --- | --- | --- |
| Created by: | Mai Quốc Thăng | Date created: |  |
| Primary actor: | BKPay system | Secondary actor: | Student |
| Description: | The BKPay system redirects the student to the login page when student wants to log out of the system or when the BKPay system encounters problem may affect the student. | | |
| Trigger: | - Student logs out of the BKPay system  - BKPay system encounters problem  - Student’s session is expired | | |
| Preconditions: | - Student already logs in the BKPay system  - BKPay system is online | | |
| Postconditions: | - BKPay system ends the student’s session  - BKPay system notifies the student. | | |
| Normal flow: | 1.0.1 Student sends a log off request  1.0.2 BKPay system receives the log off request  1.0.3 BKPay system redirects the student | | |
| Alternative flow: | 1.1.1 BKPay system encounters a problem  1.1.2 BKPay system redirects the student | | |
| Exception flows: | 1.0.2a BKPay system fails to receive the logoff request from the student  1.0.3(1.1.2) BKPay system fails to redirect the student | | |

### 2.3.3 Use case diagrams for printers managing module

| Name | SPSO views printing history of printers | | |
| --- | --- | --- | --- |
| Created by | Lê Nguyễn Gia Bảo | Date created | 28/9/2024 |
| Primary actor | SPSO | | |
| Description |  | | |
| Trigger |  | | |
| Preconditions |  | | |
| Postconditions |  | | |
| Flow |  |  |  |
| Alternative flow |  |  |  |
| Exception |  |  |  |

| Name | SPSO manages printers | | |
| --- | --- | --- | --- |
| Created by | Lê Nguyễn Gia Bảo | Date created | 28/9/2024 |
| Primary Actor | SPSO | Secondary actor | Printer |
| Description | SPSO specifies what managerial operation to perform (add printer or enable printer or disable printer) | | |
| Trigger | SPSO navigates to a page where they are allowed to manage printers | | |
| Preconditions | - SPSO’s identity has been authenticated  - SSPS System is online | | |
| Postconditions | - Configuration of concerned printers are updated | | |
| Flow | 1. SPSO chooses an action to perform (add, enable, or disable)  2. SPSO specifies the identifier of a printer  3. SPSO clicks an apply button to perform the action.  4. The action is performed and the concerned printers are notified and must respond appropriately.  5. SPSO is visually informed that the action has been performed successfully | | |
| Alternative flow |  | | |
| Exception flow: | At 2, the printer with such identifier does not exist  At 5, a system error may occur | | |

| Name | SPSO manages system configurations | | |
| --- | --- | --- | --- |
| Created by | Lê Nguyễn Gia Bảo | Date created | 28/9/2024 |
| Primary Actor | SPSO | Secondary Actor | SSPS System |
| Description |  | | |
| Trigger | SPSO navigates to a page where they are allowed to manage system configurations | | |
| Preconditions | - SPSO’s identity has been authenticated  - SSPS System is online | | |
| Postconditions |  | | |
| Flow |  | | |

| Name | SPSO generates report | | |
| --- | --- | --- | --- |
| Created by | Lê Nguyễn Gia Bảo | Date created | 28/9/2024 |
| Primary Actor | SPSO | | |
| Description | SPSO asks the system to generate monthly/yearly reports about a set of printers | | |
| Trigger | SPSO clicks a Generate Report button in the settings page. | | |
| Preconditions | - SPSO’s identity has been authenticated  - SSPS System is online | | |
| Postconditions | - SPSO can access a newly generated report file | | |
| Flow | 1. SPSO clicks a button  2. A modal dialog is presented, asking SPSO to choose between a monthly report or a yearly report. The monthly option is chosen by default.  3. SPSO clicks a button to choose one of these options, the choice is marked.  4. SPSO clicks a button to perform generation with said choice.  5. When generation is done, SPSO is visually informed that the generation has been successful. At the same time, the browser starts downloading an excel file which is the generated report.  6. SPSO clicks the Done button, or anywhere outside of the modal dialog, to finish. | | |
| Alternative flow | 2.1, 3.1, 4.1, 5.1 SPSO clicks a button, or anywhere outside of the modal dialog, to close the modal dialog thus exiting the flow. | | |
| Exception | 5.1 The generation fails due to any cause (system is offline, not enough data, etc), an error screen is shown - SPSO is visually informed that the generation has failed, accompanied by a message explaining/hinting at the cause of the error.  5.2 SPSO clicks the Done button, or anywhere outside of the modal dialog, to finish. | | |

### 2.3.4

| Name: | View personal printing log | | |
| --- | --- | --- | --- |
| Created by: | Lê Đức Nghĩa | Date created | 28/09/2024 |
| Primary actor: | Student | Secondary actor | Printer |
| Description: | Allows the student to view their personal printing log with more features such as view summary report of printed pages and view page balance | | |
| Trigger: | Student requests to view their personal log. | | |
| Preconditions: | 1. User logged in as “Student” 2. Student selects to view their personal log | | |
| Postconditions: | The system display student’s personal log | | |
| Normal flow: | 1. Student selects "View Personal Printing Log". 2. The system retrieves the student's print log. 3. Student views the summary report of printed pages 4. Student views page balance | | |
| Exception flow: | 1. If log retrieval fails, system shows an error message. | | |

| Name: | Buy additional pages | | |
| --- | --- | --- | --- |
| Created by: | Lê Đức Nghĩa | Date created | 28/09/2024 |
| Primary actor: | Student | Secondary actor | BKPay |
| Description: | Student wants to buy more additional pages as their balance is run out of pages | | |
| Trigger: | Student selects to buy more pages | | |
| Preconditions: | 1. User logged in as “Student” 2. Student selects buying feature | | |
| Postconditions: | Student successfully purchases additional page | | |
| Normal flow: | 1. Student selects "Buy Additional Pages". 2. Student choose the number of additional pages. 3. The system redirects to BKPay for payment. 4. Student completes the payment 5. The system updates the balance | | |
| Exception flow: | 1. If payment fails, the system shows an error message. 2. The balance does not update additional pages | | |

### 2.3.5 Use case diagrams for authentication module

| Name | Retrieve authentication information | | |
| --- | --- | --- | --- |
| Created by: | Lê Thanh Lâm | Date Created: |  |
| Primary actor: | Student | Second actor: | SSO |
| Description | To get the authentication information to know the role of the users | | |
| Trigger | SSO will navigate to the choosing role page after users successfully log into the page. | | |
| Preconditions: | Log in into the printing page successfully. | | |
| Postconditions: | System will redirect the user to the exact homepage which is suitable to the roles of users. | | |
| Normal flow: | 1. Users will choose one of three roles on the screen 2. SSO will check if the information of users would be appropriate for that role or not. 3. If users choose their correct role, SSO will redirect the page for users, if false, SSO will return the error and request the user opt for their role again. | | |
| Exception flow: |  | | |

| Name | Receive log in information from user | | |
| --- | --- | --- | --- |
| Created by: | Lê Thanh Lâm | Date Created: |  |
| Primary actor: | Student | Second actor: | SSO |
| Description | Checking the information of user when log-in before accessing to the Printing service | | |
| Trigger | Users will click to the “Log-in” button, and then type their username and password which have been registered before. | | |
| Preconditions: |  | | |
| Postconditions: | SSO will let the users pick their role if they successfully access the Printing service, or SSO will notify error for users. | | |
| Normal flow: | 1. Users click to the “Log-in” button 2. Users type their username 3. Users type their password 4. User click to the “Log-in” 5. SSO will then navigate to the choosing role page if successful access, or return errors if username or password typed is wrong. | | |
| Exception flow: |  | | |

1. <https://semver.org> [↑](#footnote-ref-0)