

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

1.1 Domain Context

At **Ho Chi Minh City University of Technology (HCMUT)**, printing is integral to academic life. It supports students' needs to print assignments, reports, research papers, and other academic documents. Traditional printing methods on campus often involve physically going to a printing station, submitting documents, and waiting in line for prints to be completed. This can be particularly cumbersome during busy times like exam periods or major assignment deadlines when demand for printers increases. In this context, there is a clear need for a more efficient, accessible system that provides students with better control over their printing tasks, while also allowing the university to manage its printing infrastructure and services better.

The **HCMUT Student Smart Printing Service (HCMUT_SSPS)** has been designed to meet these needs by offering a smart, centralized solution for printing across the campus. With HCMUT_SSPS, students can upload their documents remotely via a web or mobile app, select a printer on campus, and customize their printing preferences—such as paper size, number of copies, and single- or double-sided printing—without waiting at the printer. The system integrates with the university's **Single Sign-On (SSO)**, ensuring secure access and tracking of each student's print quotas. In addition, the service allows students to purchase additional pages online when they have exceeded their allocated quota, providing a seamless and user-friendly experience. This approach focuses on improving the **efficiency** of printing services, offering students more control and flexibility in how and when they print.

1.2 Stakeholders and Needs

The SSPS affects multiple entities or stakeholders, including **HCMUT students**, the **Student Printing Service Officer**, and the **University IT department**.

This system is developed for HCMUT students, so the students themselves are the center stakeholders of this system. SSPS must be easy to use, especially for low-tech students. It should provide convenience with modern features like checking printer availability, monitoring paper balance, and supporting various document types and operating systems. Students also need their transactions and documents to be secured.

Another noticeable entity using SSPS is the Student Printing Service Officer (SPSO). The SPSO needs multiple tools to monitor students' activities on the system, such as managing printers, setting the configurations, retrieving monthly and annual reports, and viewing students' printing logs.

The IT department of HCMUT will maintain the system. Implicitly, a guarantee on some levels of stability, security, and scalability will reduce the overall workload of IT staff. They also require access to the backend tools to maintain the system or back up the data.

1.3 Benefits of the System

The HCMUT Student Smart Printing Service (HCMUT-SSPS) benefits various stakeholders within the university ecosystem.

For students, it simplifies the printing process—allowing them to effortlessly upload documents, choose printers, and customize settings. Transparency is a key feature: students can easily track their printing history, promoting responsible usage. Plus, with mobile access, they can print from anywhere on campus.

The Student Printing Service Officer (SPSO) gains valuable tools as well. Centralized printer management streamlines their work, while configuration flexibility ensures the system aligns with university policies. Access to detailed usage reports empowers informed decision-making.

Finally, for the university IT department, the SSPS provides a wealth of advantages. They gain operational insights, enhance security infrastructure, learn scalability best practices, and even get some user interaction practices through support. Beyond mere printers, the SSPS contributes to an improved overall printing experience for everyone in the university community.

1.4 Functional Requirements

For HCMUT students:

- Students should be able to seamlessly upload documents via desktop or mobile device.
- The system should allow students to select the most convenient printer based on location and availability
- Students should be able to adjust settings such as paper size, color, and quality.
- The system should display a detailed log of the student's printing history, including cost and environmental impact.
- The system should support mobile access, allowing students to print documents anywhere on campus.

For Student Printing Service Officer:

- The SPSOs should be able to manage all campus printers from a single interface.
- The system should allow the SPSOs to modify settings such as quotas, access levels, and policies.
- The SPSOs must have access to detailed student usage reports, including statistics for each printer.

- The SPSOs should have real-time updates on the health, availability, and problem status of printers.
- The SPSOs should be able to control or override print queues for maintenance purposes.

For University IT Departments:

- The system should provide performance metrics and printer usage data to help IT staff improve efficiency.
- IT staff must be able to access security settings to manage user authentication and data encryption.
- The system should provide mechanisms to scale infrastructure as the student population grows.
- IT should be able to set up automated backups of critical data, including logs and configurations.
- IT staff should have tools to handle user interactions, such as troubleshooting requests and system updates.

1.5 Non- Functional Requirements

Usability:

- The system must be mobile-friendly, offering a responsive design for phones and tablets.
- The system must be easy to use, even for low-tech users, minimizing the learning curve.
- IT interfaces should be intuitive, with easy access to back-end functions.
- Transparency and Information
- The system should maintain high transparency, providing clear details on usage, paper consumption, and costs.
- The system should provide real-time data and statistics to enable informed decision-making.
- Customizable reports should be easily generated and adaptable to different timeframes (e.g. weekly, monthly).
- Real-time updates on printer availability and queue status must be provided for a smooth experience.

Security:

- Data security should protect students' documents and printing history from unauthorized access.
- Security features should prevent unauthorized access to printer settings and reports.
- The system must provide high security with built-in features to protect data integrity and privacy.
- The system must follow best practices in security, scalability, and data handling to minimize risk and overhead for IT staff.

Reliability and Availability:

- The system must maintain high reliability to avoid service interruptions and ensure smooth operation.
- Operational efficiency is key, with minimal downtime required for maintenance or system upgrades.

Scalability:

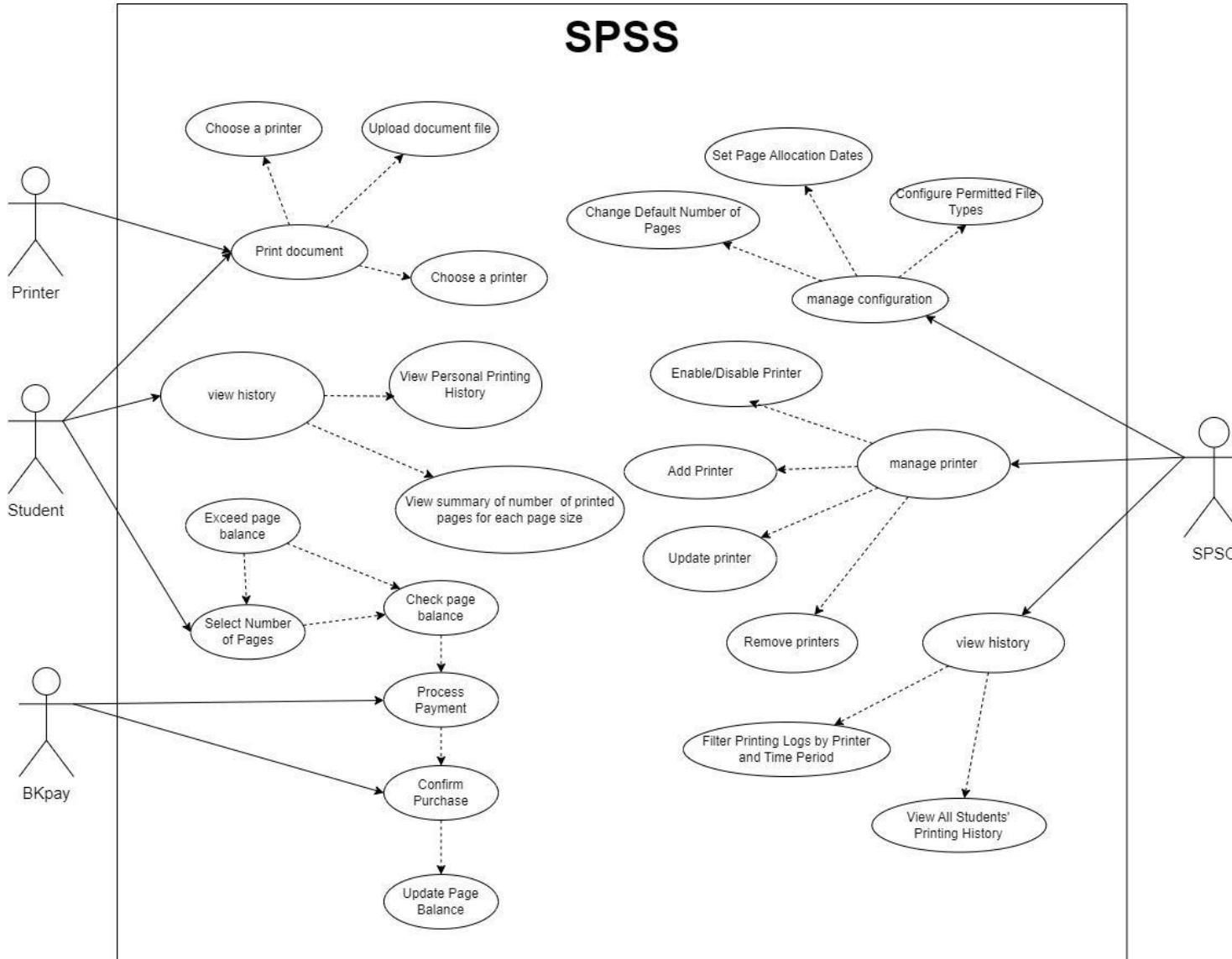
- Scalability must be flexible, allowing IT teams to expand system capacity as needed without major overhauls.
- The system should support integration with cloud services to dynamically scale computing resources and storage capacity based on real-time demand, ensuring the system can handle peak loads during high-traffic periods without any performance degradation.
- The system should be scalable across multiple university campuses or departments, allowing centralized management while accommodating local variations in printing resources and policies as the organization grows.

Customization and Flexibility:

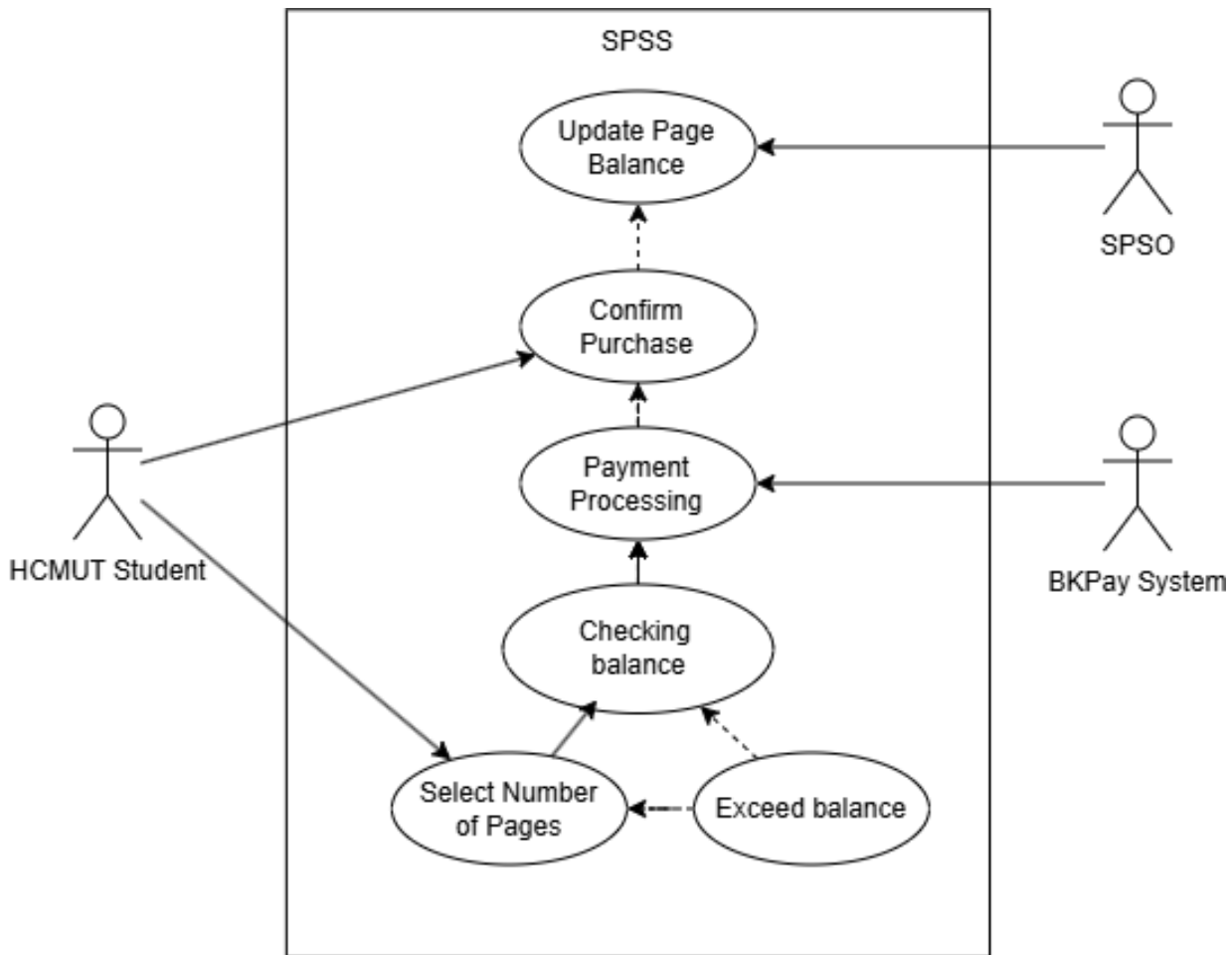
- The interface should allow for flexibility in system configuration to ensure compliance with the university's printing policy.
- The system should offer flexible user management with role-based access control, allowing administrators to customize permissions for different types of users to access specific features or reports.
- The system should allow universities to enable or disable specific modules based on their unique requirements, ensuring the solution fits different organizational needs without unnecessary complexity.

2. Use-case Diagrams (1.3)

2.1 Use-case Diagram for the Whole System



2.2 Use-case Diagram for Buying Papers Module



2.3 The Details of Use-cases in the Buying Papers Module

1. Use-case Select Number of Pages

ID and Name	BP1 Select the Number of Pages
Primary Actor	HCMUT Student
Description	The student selects the number of pages they want to purchase for printing
Trigger	The student opens the printing page and chooses a number of pages

Preconditions	The student must be logged into the system and have access to the page selection interface
Postconditions	The number of pages is selected and sorted temporarily for balance-checking
Normal Flow	<ol style="list-style-type: none"> 1. The student logs into the system. 2. The student navigates to the “Select Number of Pages” screen 3. The student inputs the number of pages 4. The system stores the selected number of pages for further processing
Alternative Flows	<ol style="list-style-type: none"> 1. If the student accidentally selects the wrong number, they can edit the selection before proceeding to payment 2. The system will allow the student to reset or modify the selection
Exceptions	<ol style="list-style-type: none"> 1. If the system can not load the selection page due to a server error, it displays an error message asking the student to retry later 2. If the selection input is invalid (e.g., negative number or zero), the system rejects the input and prompts for a valid number

2. Use-case Checking Balance

ID and Name	BP2 Checking Balance
Primary Actor	HCMUT Student
Description	The system checks whether the student’s balance is sufficient to cover the selected number of pages
Trigger	The student selects the “Proceed to Payment” option after selecting pages.
Preconditions	The student has already selected the number of pages

Postconditions	The system compares the page selection against the student's balance and proceeds with the next step based on the result
Normal Flow	<ol style="list-style-type: none"> 1. The system retrieves the student's current balance from the database 2. The system checks if the balance is sufficient to cover 3. If the balance is sufficient, the system allows the student to proceed to payment
Alternative Flows	<ol style="list-style-type: none"> 1. If the balance is insufficient, the system notifies the student that they need to reduce the number of pages or top up their balance 2. The student is given the option to revise their selection or continue to the next step.
Exceptions	<ol style="list-style-type: none"> 1. If the system fails to retrieve the balance due to a network error, it shows a message asking the student to try again 2. If the database is down, the system will notify the student that balance checking is unavailable

3. Usecase Exceed Balance

ID and Name	BP3 Exceed Balance
Primary Actor	SPSS
Description	The system detects that the number of pages selected by the student exceeds their current balance, preventing them from proceeding to payment
Trigger	The student attempts to confirm the selection, but the system detects an insufficient balance
Preconditions	The student has selected a number of pages, and the system checks the balance
Postconditions	The student is notified of the insufficient balance and offered the option to reduce the number of pages or add more funds to proceed

Normal Flow	<ol style="list-style-type: none"> 1. The system checks the balance after the student selects a number of pages 2. The system determines that the balance is insufficient 3. The system notifies the student of the insufficient balance and offers options to either reduce the number of pages or add more funds to proceed
Alternative Flows	<ol style="list-style-type: none"> 1. The student chooses to increase their balance instead of reducing the number of pages 2. The system redirects the student to a balance top-up interface
Exceptions	<ol style="list-style-type: none"> 1. If the balance check fails due to a system error, the system displays an error message and halts the process 2. The student is notified to try again later if the error persists.

4. Usecase Payment Processing

ID and Name	BP4 Payment Processing
Primary Actor	BKPay System
Description	The system processes the payment with the BKPay system
Trigger	The system confirms the sufficient balance and initiates the payment process
Preconditions	The system has already verified that the student's balance is sufficient to complete the purchase
Postconditions	The payment is initiated, we are waiting for student confirmation
Normal Flow	<ol style="list-style-type: none"> 1. The system confirms the sufficient balance and initiates the payment process 2. The system forwards the payment request to BKPay 3. BKPay asks the student to confirm the payment

Alternative Flows	If the student chooses to cancel the payment, the system reverts the process and returns to the selection step
Exceptions	If BKPay fails to process the payment due to an external issue, the system displays an error and tries again after a short delay

5. Usecase Confirm Purchase

ID and Name	BP5 Confirm Purchase
Primary Actor	HCMUT Student
Description	The system confirms the student's purchase and updates their page balance accordingly
Trigger	The system receives confirmation from the student that the payment was successfully initiated
Preconditions	The student's payment has been successfully initiated
Postconditions	The student's confirmation page is displayed
Normal Flow	<ol style="list-style-type: none"> 1. The system receives a payment initiate success message from BKPay 2. The system notifies the SPSO to update the student's page balance 3. The system displays a success message to the student, showing the new page balance
Alternative Flows	If the student chooses to cancel the payment, the system reverts the process and returns to the selection step
Exceptions	If the SPSO system is down, the balance is not updated, and the system logs the issue

6. Usecase Update Page Balance

ID and Name	BP6 Update Page Balance
Primary Actor	SPSO
Description	The SPSO system updates the student's page balance after a successful payment
Trigger	The system receives confirmation from the HCMUT student and successful payment from BKPay
Preconditions	The payment has been successfully processed and confirmed by the BKPay system
Postconditions	The student's page balance is updated to reflect the last purchased pages
Normal Flow	<ol style="list-style-type: none"> 1. The system receives a notification of a successful payment from BKPay 2. The system notifies the SPSO system to update the student's page balance 3. SPSO updates the student page balance with the last purchased pages 4. The updated page balance is displayed to the student
Alternative Flows	<ol style="list-style-type: none"> 1. If the update is delayed, the system tries again to update the page balance after a short interval 2. The student may see a "Balance update pending" message if the update takes longer than expected
Exceptions	<ol style="list-style-type: none"> 1. If the SPSO system is down or fails to update the balance, the system logs the issue and notifies the student 2. The student is informed of an issue with their balance update and may need to contact support if the problem continues to happen