Software Requirements Specification

for

A SMART PRINTING SERVICE FOR STUDENTS AT HCMUT

Version 1.0 approved

Prepared by:

1. **Nguyễn Đức Tâm – 2252734**
2. **Nguyễn Dương Khánh Tâm – 2252733**
3. **Nguyễn Đoàn Hải Băng – 2252078**
4. **Nguyễn Bình Nguyên – 2252545**
5. **Nguyễn Bá Vương – 2252922**

Department of Software Engineering
Faculty of Computer Science and Engineering
Ho Chi Minh City University of Technology – VNU-HCM

09/22/2024

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Nguyễn Đức Tâm	19/09/2024	Add 1.2 Stakeholders and Needs.	0.1
Nguyễn Dương Khánh Tâm	21/09/2024	Add 1.1 Domain context.	0.2
Nguyễn Đoàn Hải Băng	21/09/2024	Add 1.3 Benefits of the system.	0.3
Nguyễn Bá Vương	21/09/2024	Add 1.5 Non- Functional Requirements.	0.4
Nguyễn Bình Nguyên	21/09/2024	Add 1.4 Functional Requirements.	0.5
Nguyễn Đức Tâm	28/09/2024	Add 2.1, 2.2 and 2.3.4 Use Case: Print documents.	0.6
Nguyễn Bình Nguyên	29/09/2024	Add 2.3.2 Use Case: Choose a printer.	0.7
Nguyễn Dương Khánh Tâm	29/09/2024	Add 2.3.3 Use Case: Specify printing properties.	0.8
Nguyễn Đoàn Hải Băng	29/09/2024	Add 2.3.1 Use Case: Upload document files.	0.9
Nguyễn Bá Vương	29/09/2024	Add 2.3.5 Use Case: Log activity.	1.0
Nguyễn Đoàn Hải Băng	08/11/2024	Add 4.1. Layered architecture diagram	1.1

Nguyễn Bình Nguyên	09/11/2024	Add 4.1.1. Presentation Strategy and 4.1.3. API management	1.2
Nguyễn Bá Vương	09/11/2024	Add 4.1.2. Data storage	1.3
Nguyễn Đức Tâm	10/11/2024	Add 4.2 Component diagram	1.4
Nguyễn Dương Khánh Tâm	17/11/2024	Add 4.2 Component diagram description	1.5

1. Requirement elicitation

1.1 Domain context

Despite the ongoing digitalization of learning materials, printing still plays a crucial role in the academic life of university students, serving essential functions for both students and lecturers. Indeed, a survey conducted by National Association of College Stores revealed that 68% of students remain reliant on printed lecture notes, handouts, and textbooks for effective engagement with course materials, as physical copies enhance comprehension and retention. Additionally, according to the International Society for Technology in Education, 65% of university professors compel assignments and reports to be submitted in printed form and adhere to certain formatting specification, and a research from The Reading and Writing Quarterly indicates that students achieve a 30% improvement in academic performance when they print drafts for peer reviews and provide personal editing for revisions. On top of that, printing study notes, mind maps, and resumes is vital for students preparing for exams or job applications. Therefore, it can be inferred that revolutionizing printing services for students and lectures alike will substantially contribute to their academic success.

Given the critical role of printing in students' academic lives, the HCMUT Student Smart Printing Service (SSPS) is designed to provide a seamless and efficient printing solution tailored to their diverse needs across university campuses. This service streamlines the printing process by enabling students to upload documents, choose specific printing options, and select from various available printers. Furthermore, the SSPS assists students in managing their printing quotas each semester and offers the convenience of purchasing additional pages through an integrated online payment system. Due to web-based access, the HCMUT_SSPS ensures that students can effortlessly print documents, track their printing usage, and handle payments, making it a vital resource for enhancing their academic experience.

1.2 Stakeholders and Needs

In the HCMUT Smart Printing Service (HCMUT_SSPS), several stakeholders have distinct needs. There are namely 5 types of stakeholders currently being considered in the system: The Students who are the end-users, the Student Printing Service Officers (SPSO) and the IT support center both being the system managers, the system owner is the University administration itself and the external stakeholder is the Online payment system (BKPay).

The **Students** require a seamless interface for uploading documents, selecting printers, setting printing preferences, managing their page balance, and viewing their printing history. In addition, the students need access to the printing service across the campus. They also need the option to buy additional printing pages via an integrated online payment system.

Student Printing Service Officers (SPSOs) need robust administrative controls to manage printer settings, configure the system, view and filter detailed logs, adjust default student printing quotas, configure accepted file formats, and receive usage reports each month and each year.

IT Support Center is responsible for maintaining system performance, ensuring smooth integration with HCMUT's authentication service (HCMUT_SSO), and resolving technical issues. Moreover, they have to ensure the web-based app is working as intended.

University administration requires periodic reports on system usage, including monthly and yearly summaries, to assess resource allocation and budgetary impact.

Finally, the **BKPay online payment system** act as a payment gateway and must securely facilitate transactions for students purchasing additional pages, ensuring proper

integration and communication with the printing service, and storing the transaction records accurately.

1.3 Benefits of the system

The students, as primary users, gain significant convenience and flexibility. The system provides an intuitive interface for uploading documents, selecting printing preferences, and managing their page balance. By allowing access to multiple printers across the campus, students can print documents easily from any location. Furthermore, the integration with the BKPay system enables seamless transactions for purchasing additional pages, ensuring they can manage their printing needs efficiently. Access to a personal printing log allows students to track their usage, promoting responsible and informed use of their allocated resources.

For Student Printing Service Officers (SPSOs), the system offers advanced administrative controls that enhance operational efficiency. They can manage printer settings, configure system parameters, and adjust student printing quotas. The ability to view and filter detailed logs provides insights into printing activities, while monthly and yearly usage reports aid in planning and decision-making. Additionally, the capacity to configure accepted file formats ensures the system remains flexible and meets evolving needs, allowing SPSOs to maintain a high level of service quality.

The IT Support Center benefits from a streamlined maintenance and support process. Ensuring smooth integration with the HCMUT_SSO authentication service, they can address technical issues promptly, maintaining system reliability. The web-based and mobile apps are designed for optimal performance, minimizing downtime and enhancing the user experience. This proactive approach to system management helps prevent disruptions and ensures continuous service availability.

For the university administration, the system provides comprehensive insights into resource utilization and budgeting. Regularly generated reports on system usage, including monthly and yearly summaries, allow for informed assessments of resource allocation and

financial impact. These insights can guide strategic decisions to optimize the university's printing services and support sustainable growth.

Lastly, the BKPay online payment system benefits from a secure and efficient transaction process. By facilitating seamless payments for additional printing pages, BKPay ensures transactions are processed accurately and securely. Proper integration with the printing service enhances user trust and satisfaction, reinforcing the payment system's reliability and effectiveness. This collaboration not only supports students' needs but also strengthens the university's financial infrastructure.

1.4 Functional Requirements

1.4.1. Students

1. Students must be able to upload document files to the system for printing.
2. Students should be able to choose a specific printer by location, model, or description.
3. Students can specify the printing properties, including paper size (A3, A4), number of copies, one-/double-sided printing, and specific pages to print.
4. Students should be able to view their remaining page balance (default and purchased) and ensure that their requested print job does not exceed their balance.
5. Students can view their printing logs for a specified time period, including a breakdown of the number of pages printed by page size.

1.4.2. Student Printing Service Officers (SPSO)

1. SPSO can add, enable, and disable printers in the system.
2. SPSO should be able to access and filter student printing history logs by student, printer, or time period (date-to-date).

3. SPSO can configure the default number of A4 pages allocated to students each semester and set the equivalent ratio of A3 pages to A4.
4. SPSO can configure and update the file types that are accepted for printing.
5. SPSO should be able to view automatically generated reports on system usage at the end of each month and year.

1.4.3. IT Support Center

1. The website should provide an administrative dashboard where the IT Support Center can monitor real-time system health
2. The website must have error logging capabilities that record system issues (e.g., failed print jobs, server errors) and generate real-time alerts for the IT Support Center
3. The website should provide detailed logs of user authentication activity (via HCMUT_SSO), allowing IT to monitor login attempts, successful logins, and any unauthorized access attempts.
4. The website should allow IT to remotely configure printer settings
5. The website should provide IT with the ability to create, modify, or revoke user roles and permissions, ensuring the right individuals have access to the correct features based on their responsibilities within the system.

1.4.4. University Administration

1. The administration should be able to view reports on printing system usage, including monthly and yearly data.
2. The website should provide the University Administration with reports that track the environmental impact of the printing service, including paper usage, energy consumption
3. The website must allow the University Administration to configure and send automated notifications to students and staff about policy updates

4. The website must notify University Administration when printers require maintenance or replacement
5. The website must allow University Administration to gather and review feedback from students and staff regarding their experience with the printing service

1.4.5. Online Payment System (BKPay)

1. BKPay must process payments for students who want to buy additional printing pages.
2. BKPay should store and provide transaction records for students and administrators to view.
3. In case of technical issues or failed transactions, BKPay must provide a method for refunds or adjustments.
4. Ensure that all transactions are securely processed, and students' payment information is safeguarded.
5. BKPay must send real-time payment confirmation and notifications to students and the printing system upon successful transactions

1.5 Non- Functional Requirements

- Performance

Concurrent Users:

Requirement: The system must support up to 500 concurrent users without performance degradation.

Example: During peak hours, such as exam periods, up to 500 students might be using the printing service at the same time. The system should handle this load without slowing down or crashing.

Maximum Upload Load:

Requirement: The system must allow users to upload up to 1GB at a time.

Example: When a student wants to upload many 10MB PDF files, the system should handle all the files at the same time up to 1GB of memory to ensure a smooth user experience.

Report Generation Time:

Requirement: The system must generate reports within 10 seconds.

Example: When the SPSO requests a monthly usage report, the system should compile and display the report within 10 seconds.

Printer Support:

Requirement: The system shall support and interact with up to 80% of printers currently being used across all campuses.

Example: The university might have 100 printers distributed across various buildings and campuses. The system should manage and coordinate most of these printers efficiently.

- Security

User Authentication:

Requirement: The system must authenticate all users via the HCMUT_SSO authentication service.

Example: Before accessing the printing service, students must log in using their university credentials through the HCMUT_SSO system.

Encrypted Storage:

Requirement: The system must store all sensitive data, such as payment information, in an encrypted format.

Example: Payment details entered by students for buying additional pages should be stored in an encrypted database to protect against data breaches.

Access Logging:

Requirement: The system must log all access and actions for auditing purposes.

Example: Every time a student prints a document or an SPSO changes a setting, the system should record these actions in a log for future reference.

Role-Based Access Control:

Requirement: The system must enforce role-based access control for different user roles (students, SPSO).

Example: Students should only have access to their own printing history, while the SPSO should have access to all students' logs and system configurations.

- Usability

User-Friendly Interface:

Requirement: The system must provide a user-friendly interface accessible via web-based application

Example: The interface should be intuitive, with clear navigation menus and easy-to-use forms for uploading documents and selecting printing options.

Clear Error Messages:

Requirement: The system must provide clear error messages and guidance for resolving issues.

Example: If a student tries to upload an unsupported file type, the system should display a message like "Unsupported file type. Please upload a PDF or DOCX file."

- Reliability

High Uptime:

Requirement: The system must have an uptime of 99.9% per month.

Example: The system should be available and operational for all but approximately 43 minutes per month, ensuring minimal downtime.

Daily Backups:

Requirement: The system must back up all data daily and store backups for at least 30 days.

Example: Every night, the system should create a backup of all data, which can be restored if needed, and keep these backups for a month.

Data Consistency:

Requirement: The system must ensure data consistency across all components.

Example: If a student buys additional pages, their new balance should be immediately reflected in all parts of the system, including their dashboard and the printing service.

Maintainability

Easy Printer Management:

Requirement: The system must allow for easy addition and removal of printers by the SPSO.

Example: The SPSO should be able to add a new printer to the system or disable an existing one through a simple interface.

Scalability

Horizontal Scaling:

Requirement: The system must be able to scale horizontally to accommodate increasing numbers of users and printers.

Example: If the number of students using the service doubles, the system should be able to add more servers to handle the increased load without significant changes to the existing infrastructure.

Interoperability

Integration with Existing IT Infrastructure:

Requirement: The system must integrate seamlessly with the university's existing IT infrastructure, including the HCMUT_SSO authentication service and BKPay payment system.

Example: When a student logs in, the system should authenticate them using the HCMUT_SSO service and allow them to make payments through BKPay without any issues.

Compatibility with Operating Systems and Browser:

Requirement: The system must be accessed through a website on Google.

Example: The system should work smoothly when open Google Chrome on Windows, macOS, Linux,…

Availability

24/7 Availability:

Requirement: The system must be available 24/7, with scheduled maintenance windows communicated in advance to users.

Example: The system should be operational at all times, except during scheduled maintenance, which should be announced to users at least 24 hours in advance.

Real-Time Status Updates:

Requirement: The system must provide real-time status updates on printer availability and system health.

Example: If a printer is out of paper or experiencing issues, the system should display this information in real-time to prevent students from selecting that printer.

Environmental Impact

Optimized Printing Processes:

Requirement: The system must optimize printing processes to reduce paper and ink usage.

Example: The system allow users to preview documents before printing to avoid unnecessary prints.

Eco-Friendly Printing Options:

Requirement: The system must provide users with options to print in eco-friendly modes, such as double-sided printing.

Example: When selecting printing options, students should be prompted to choose double-sided printing to save paper.

2. Use-case Diagrams

2.1 Use-case Diagram for the Whole System

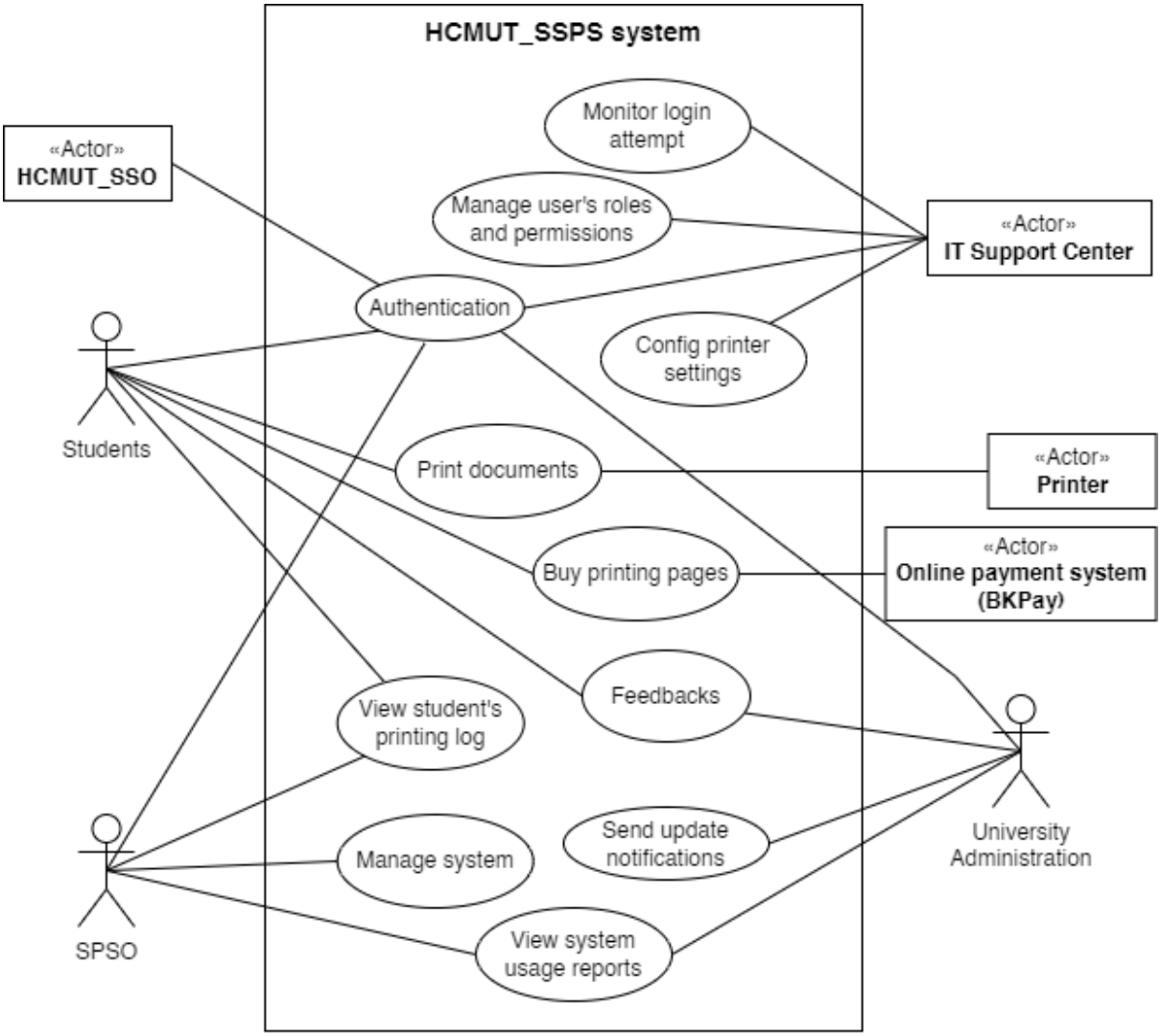


Image 1. The HCMUT_SSPS Use-case Diagram

2.2 Use-case Diagram for “Print documents” Module

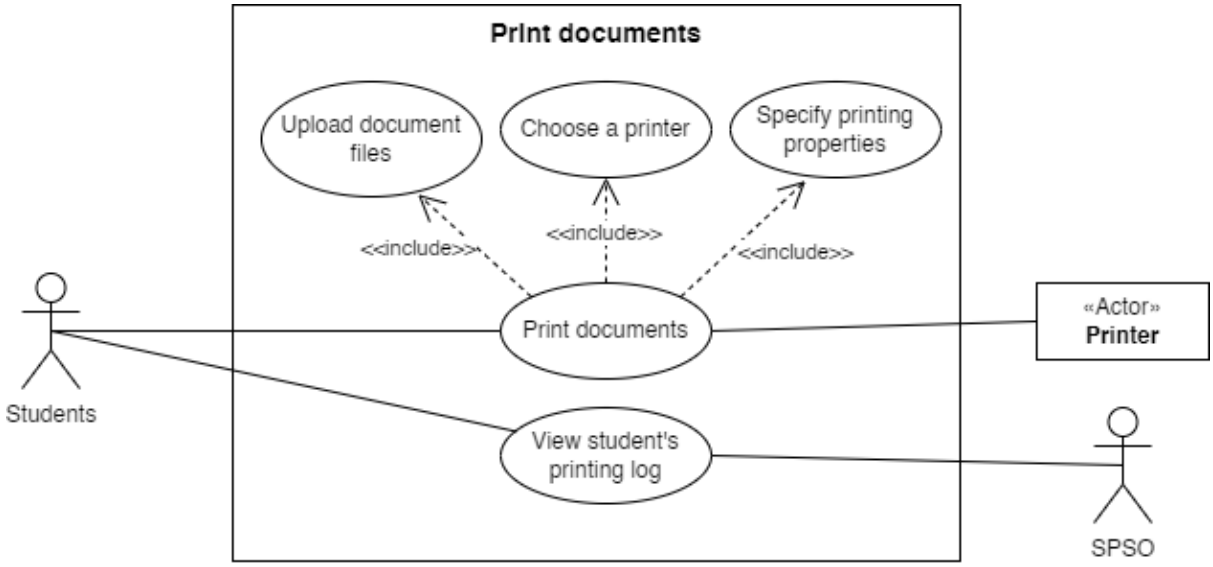


Image 2: The “Print documents” module Use-case Diagram

2.3 The Details of Usecases in “Print documents” Module

2.3.1 Usecase: Upload document files

ID and Name	UC-01 Upload document files	
Created by	Nguyễn Đoàn Hải Băng	Date created: 09/29/2024
Primary actor	Students	Secondary actor: Printer
Description	The student uploads a document file to the system in preparation for printing.	
Trigger	Students indicate that they want to upload files to be printed.	
Preconditions	PRE-1: The student’s device has connected to the Internet. PRE-2: The student is logged in to the system. PRE-3: The account has been assigned with “Student” role.	
Postconditions	POST-1: The document file is successfully uploaded to the system.	

	POST-2: The document is ready for selection of printer and print properties.
Normal flow	**1.0: Upload document file to the system** 1. The student navigates to the "Upload Document" section. 2. The system displays the upload interface. 3. The student clicks the upload button and selects a document from their device. 4. The system uploads the document. 5. The system confirms that the file is ready for printing
Alternative flow	**1.1: The file size exceeds the allowed limit** 4a. The system notifies the student about the excessive file size 4a1. The system cancels the upload. 4a2. The system asks students to upload other files. *The Use Case continue from step 3.*
Exception	**1.1.E1: The network connection is lost during upload** 3a. If the student manages to reconnect. 3a1. The system notifies the student about the network connection loss. 3a2. The system asks the student to reupload the file. *The Use Case continues from step 3.* 3b. If the student disconnects entirely. 3b1. The system notifies the student about the network connection loss. *The Use Case ends.*

2.3.2 Usecase: Choose a printer

ID and Name	UC-02 Choose a printer	
Created by	Nguyễn Bình Nguyên	Date created: 09/29/2024
Primary actor	Students	Secondary actor: Printer

Description	Student specifies the printer that they want to use by its ID or its brand/manufacturer name, printer model, short description, and the location. The system will give the permission to use the printer that students chose.
Trigger	Students indicate that they want to choose a printer after uploading document files onto the system.
Preconditions	PRE-1: Student's identity has been authenticated. PRE-2: The student has already uploaded the document files. PRE-3: There must be at least 1 printer ready to be used.
Postconditions	POST-1: The request was sent to the corresponding printer. POST-2: The student is shown a confirmation screen or message with the chosen printer details before proceeding with printing. POST-3: The printer's resources information (ID, number of pages, amount of ink) are sent to the user.
Normal flow	**2.0: Choose a printer from a list of printers** 1. The student uploads the document files. 2. The system displays a list of printers 3. The student selects a desired printer 4. The system stores the request and notify the student.
Alternative flow	**2.1: Choose a printer that is not available** 3a. The student selects an unavailable printer from the list. 3a1. The system alerts the student that the selected printer is currently being used or under maintenance. 3a2. The student selects an alternative printer then the system verifies its availability. *The Use Case continues from step 3.*
Exception	**2.1.E1: All printers are not currently available** 2a. The system detects that no printers are available. 2a1. The system displays an error message to the student. 2a2. The student can refresh the printer list in case any printer becomes available or they cancel the printing job and exit the process. 2a3. If the student finds an available printer, the system

	proceeds to step 2 of the normal flow.

2.3.3 Usecase: Specify Printing Properties

ID and Name	UC-03 Specify Printing Properties	
Created by	Nguyễn Dương Khánh Tâm	Date created: 09/29/2024
Primary actor	Students	Secondary actor: Printer
Description	Student specifies the printing properties such as paper size, pages (of the file) to be printed, single-/double-sided printing, number of copies, etc. The system will then configure the permitted file types, as specified by the Student Printing Service Officer (SPSO), and record the printing properties.`	
Trigger	Students indicate that they want to specify printing properties after choosing a printer.	
Preconditions	PRE-1: Student's identity has been authenticated. PRE-2: The student has already uploaded the document files. PRE-3: The student has chosen a printer. PRE-4: The permitted file types are configured by the SPSO.	
Postconditions	POST-1: The printing properties are set as per the student's specifications. POST-2: The system checks for file type compliance and sends the request to the printer. POST-3: The student is shown a confirmation screen or message with the chosen printing properties before proceeding with printing.	

Normal flow	**3.0: Specify valid file settings** 1. The system displays options to specify printing properties. 2. The student specifies the printing properties (e.g., paper size, pages to be printed, single-/double-sided, number of copies). 3. The system verifies the file type against the configured permitted file types. 4. The system checks if the selected printer has enough resources. 5. The system stores the request and notifies the student of successful submission.
Alternative flow	**3.1: Specify invalid file types** 3a. The student uploads a document with a file type that is not permitted. 3a1.The system alerts the student that the file type is not supported. 3a2. The student returns to Use Case "UC-01 Upload document files" to upload a permitted file type *The Use Case continues from step 4.*
Exception	**3.1.E1: Insufficient printer resources** 4a. The system detects that the selected printer does not have enough resources (e.g., paper, ink) to complete the printing job for the specified printing properties. 4a1. The system displays an error message to the student indicating the specific resource shortage. 4a2. The student can choose to: a) Return to Use Case "UC-02 Choose a printer" to select another printer from the list. b) Cancel the printing process.

1.1.1 Usecase: Print documents

ID and Name	UC-04 Print documents

Created by	Nguyễn Đức Tâm	Date created: 09/28/2024
Primary actor	Students	Secondary actor: Printer, Online payment service (BKPay)
Description	The process of a student printing the uploaded documents from the printing system after he had chosen a printer and specified the printing properties.	
Trigger	Students indicate that they want to print documents after having uploaded the documents, chosen the printer and specified the printing properties.	
Preconditions	PRE-1: Student's identity has been authenticated. PRE-2: The student has uploaded at least one document file. PRE-3: The student has selected a printer. PRE-4: The student has specified the printing properties.	
Postconditions	POST-1: The documents are successfully printed according to the specified printing properties. POST-2: The student is shown a confirmation screen. POST-3: The system asks if the student wants to print again. POST-4: The printing action is recorded by the system.	
Normal flow	**4.0: Printing the documents with the selected printer** 1. The student can select the specific printing time at their disposal. 2. The student confirms the file settings and the information of the designated printer. 3. The student presses the "Print" button. 4. The system checks for the student's remaining pages. 5. The system sends the printing requirements to the selected printer. 6. The printer starts printing and sends a notification to the student indicating that it has done the printing process. 7. The system logs the printing activity.	
Alternative flow	**4.1: Insufficient student's remaining pages** 4a. The student runs out of the default pages	

	4a1.The system alerts the student that there is not enough remaining pages and ask if they want to buy more pages. 4a2. The student purchases the pages using the online payment service. *The Use Case continues from step 5.*
Exception	**4.1.E1: Failed transaction when purchasing pages.** 4a. The student tries to buy the pages from the BKPay but ran into an error when accessing the BKPay page. 4a1. The system asks the student to retry the payment process or cancel the print job. *The Use Case ends and return to homepage.*

1.1.2 Usecase: View student's printing logs

ID and Name	UC-05 View student's printing logs	
Created by	Nguyễn Bá Vương	Date created: 09/28/2024
Primary actor	Students	Secondary actor: SPSO
Description	The students can view their own printing history. In addition, SPSO can also view the student's printing logs.	
Trigger	The student or the SPSO indicates that they want to view the printing logs.	
Preconditions	PRE-1: Student/SPSO's device is connected to the Internet. PRE-2: Student/SPSO must be authenticated by the system. PRE-3: The account has been assigned with "Student" or "SPSO" role.	
Postconditions	POST-1: The system returns a table with different logs entry including the file's name, date of printing, the printer's code, the number of page printed and the number of times printed for the student/SPSO to see.	

	POST-2: The student/SPSO has finished checking the their own/student's printing logs.
Normal flow	**5.0: The student's view of the printing logs** 1. The user navigates to the "Lịch sử in ấn" page. 2. The system shows a table of the printing history of the student including the file's name, the date of when printing finished, the printer's ID, the number of page printed and the number of times printed. 3. The user can select a page by clicking the page number if the printing logs are too long. 4. The user selects a printing log entry they want to view. 5. The system shows the information in more details including the data in step 2 plus the date of when the user entered the specifications for printing, the page's size and the number of the page's sides.
Alternative flow	**5.1: The SPSO's view of the student's printing logs** 1a. The user navigates to the "Lịch sử in ấn" page and selects a specific student's ID. *The Use Case continues from step 2.*
Exception	**4.1.E1: Error when specifying the studen't ID.** 1a. The user tries to type in the student's ID but ran into an error due to wrong ID format or that ID doesn't exist. 1a1. The system asks the user to type again. *The Use Case continues from step 1.*

2. System modelling

2.4 Activity Diagram

2.4.1 Usecase: Upload document files

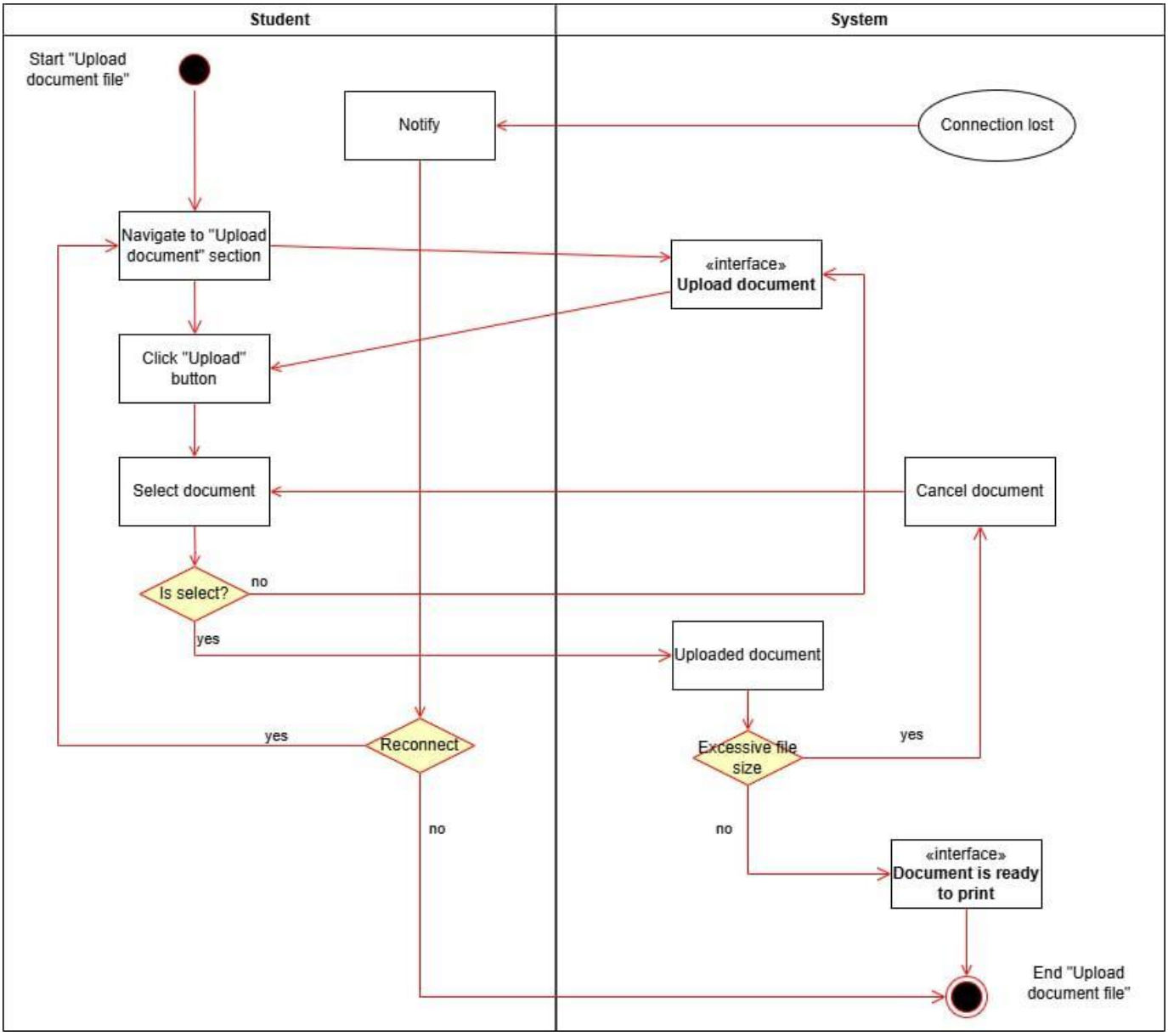


Image 3: The "Upload document files" Activity Diagram

The diagram illustrates the process for students to upload document files for printing, featuring distinct sections for "**Student**" and "**System**". The sequence begins with the **Student** navigating to the "Upload document" section, it will turn to the "Upload document" interface. Then the **Student** clicks the "Upload button" and selects the document to upload. If the **Student** selects a document, the document will be uploaded to the **System**. Otherwise, the **System** will return to the "Uploaded document" interface. Next, the **System** checks the file size of the uploaded document. If the document's size is

acceptable, the **System** will display that the document is ready to print and the activity will end. Otherwise, the **System** canceled the document and ask the user to select another document. During the activity, if the user occurs “Connection lost”, the **System** will notify and ask them to reconnect. If they reconnect, they will have to navigate to the “Upload document” section and continue. Otherwise, end the upload document activity.

2.4.2 Usecase: Choose a printer

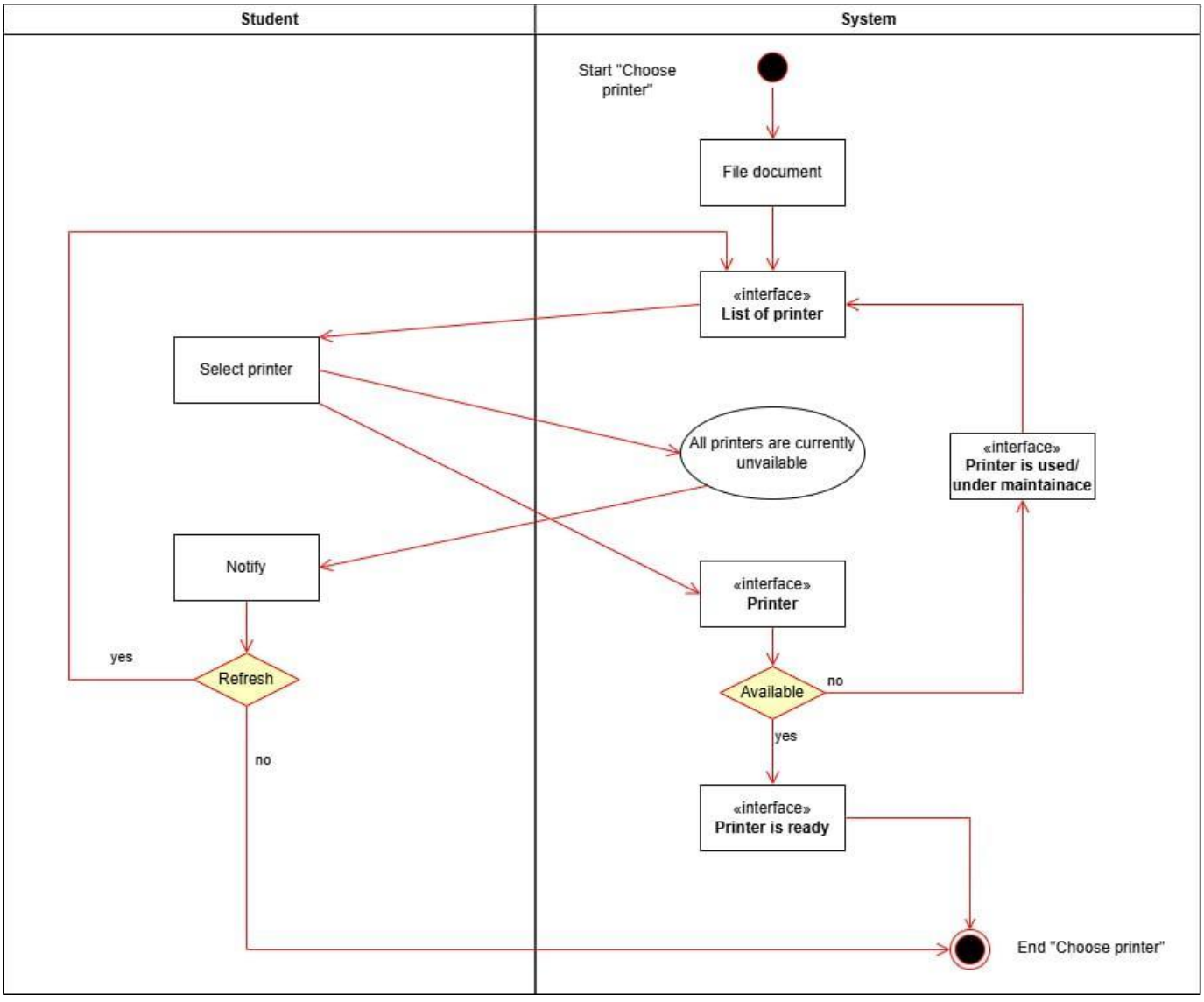


Image 4: The “Choose a printer” Activity Diagram

The activity diagram demonstrates the process of choosing a printer by a student and the corresponding interactions with the system. The diagram is divided into two swimlanes: "**Student**" and "**System**". The sequence begins with the current uploaded document and displays the “List of printers” interface. Then, the **Student** needs to select a printer. In the case that all of the printers are currently unavailable, the **System** will notify the user and ask them to refresh the page. If the **Student** chooses to refresh the page, the **System** will display the “List of printers” interface again with updated printer information (the user can refresh the page until they can select a printer). Otherwise, end “Choose printer”. When the **Student** selects a printer, the printer interface will appear. The **System** checks the printer if it is available or not. If the printer is available, the **System** will display the “Printer is ready” prompt and end “Choose printer”. On the other hand, the **System** will display the “Printer is used/ under maintenance” prompt and return to the “List of printer” interface.

2.4.3 Usecase: Specify Printing Properties

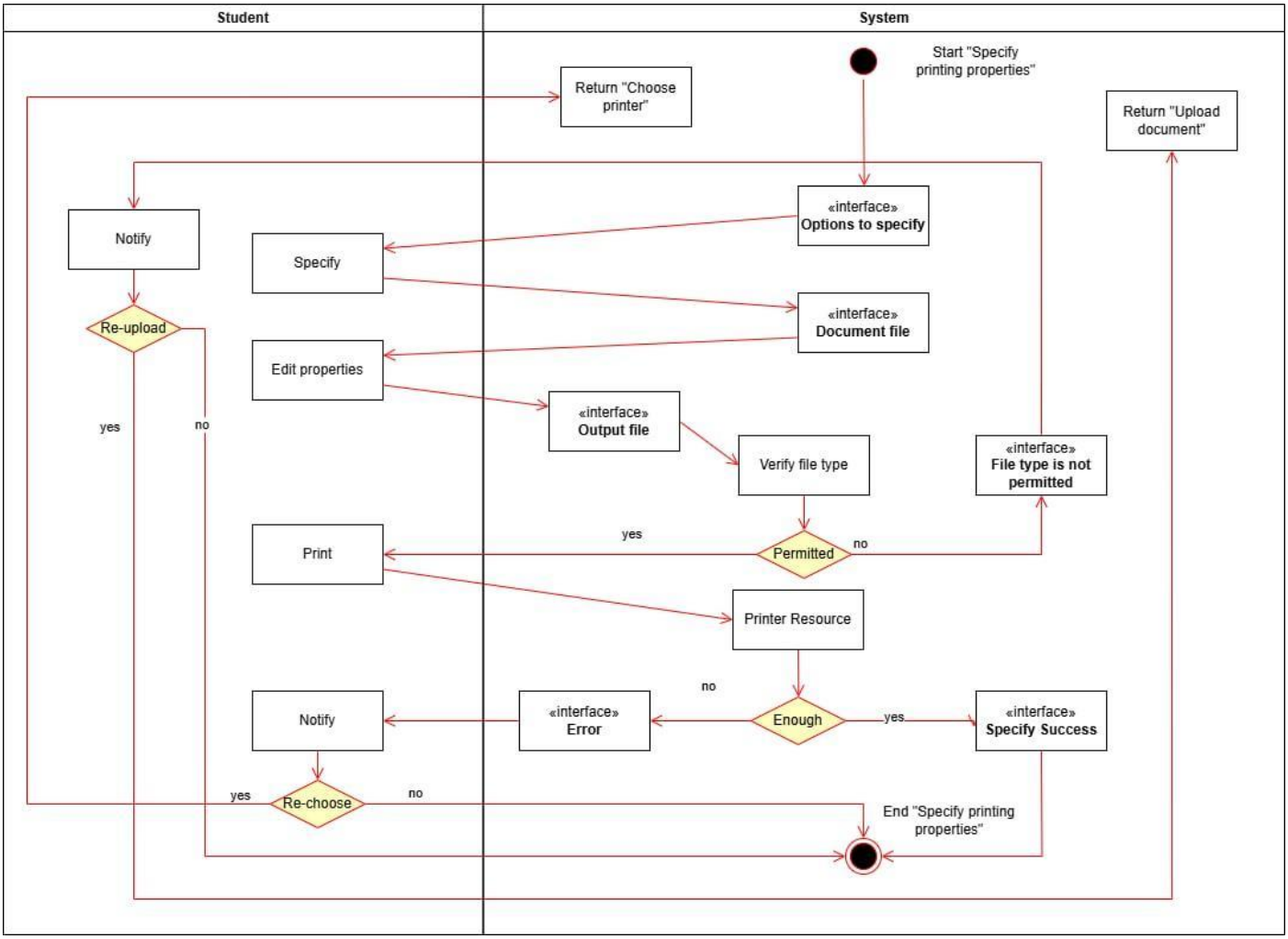


Image 5: The "Specify Printing Properties" Activity Diagram

The diagram outlines the process of specifying printing properties and is divided into two sections: "**Student**" and "**System**." The sequence starts with the "Options to specify" interface. Then, the **Student** chooses to specify the document. After that, the "Uploaded document" interface appears. Next, the **Student** edits the properties of the document. When the **Student** finishes editing the properties, the "Output file" interface appears and the **System** considers whether the file type is permitted by the **SPSO** or not. If the file type is permitted, the user needs to confirm the document's settings. If the file type is not permitted, the "File type is permitted" interface is displayed and the **System** notifies the **Student**. When the **Student** chooses to re-upload the document, return to the "Upload document" step. Otherwise, end "Specify printing properties". After the **Student**

has pressed the “Print” button, the **System** will consider whether the printer resources are sufficient enough or not. If the printer has sufficient resources, the **System** will display “Specify Success” and end “Specify printing properties”. Otherwise, the **System** will display “Error” and ask the **Student** to choose another printer. If the **Student** wants to re-choose the printer, return to the “Choose printer” step. Otherwise, end “Specify printing properties”.

2.4.4 Usecase: Print documents

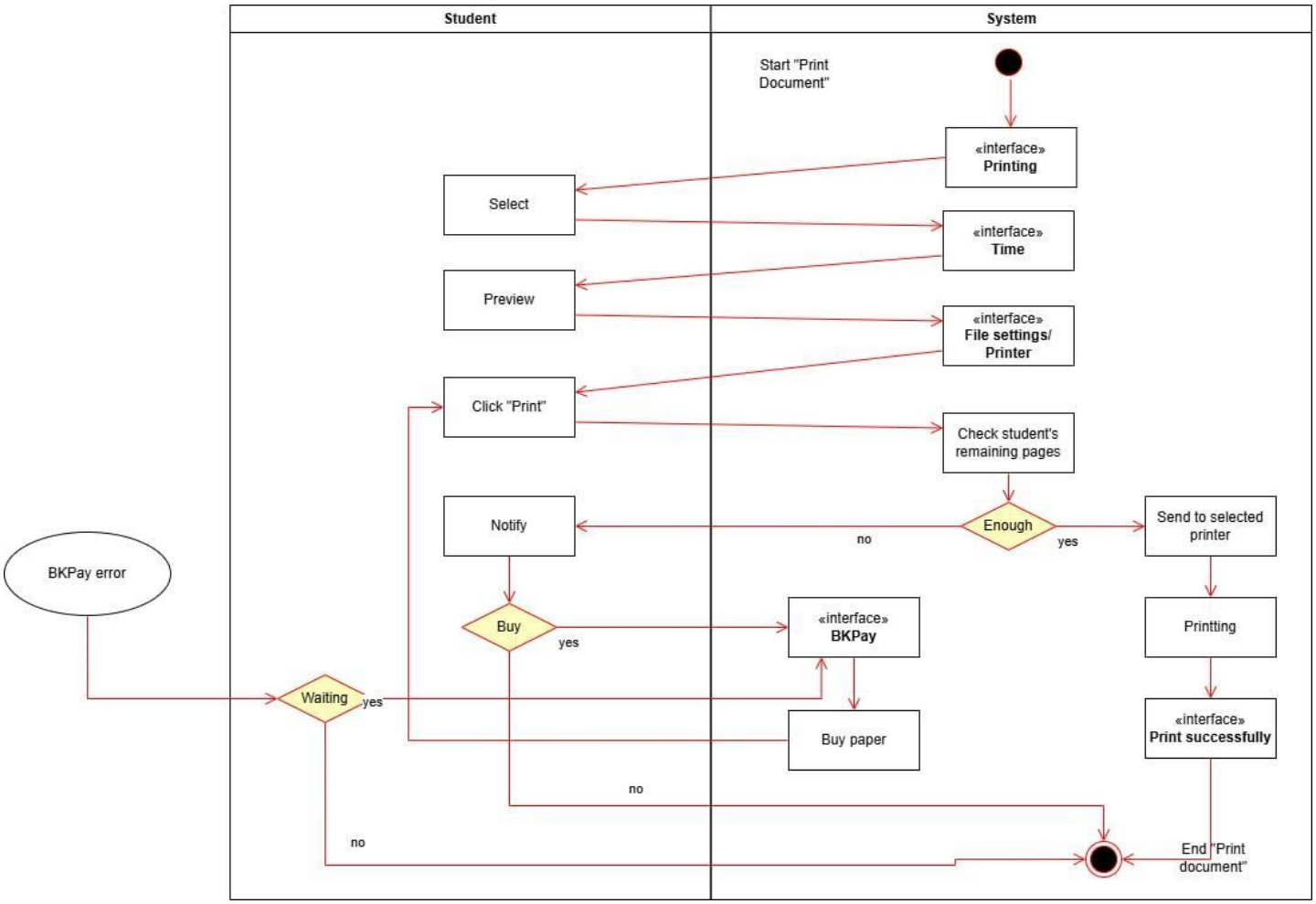


Image 6: The “Print documents” Activity Diagram

The diagram depicts the process of a student printing a document using a system. The diagram is divided into two sections: "**Student**" and "**System**". The process begins

with the “Printing” interface. The **Student** can select the time of printing and the **System** will display it to the user. Next, the **Student** can preview the uploaded document and the **System** will show the file settings and the chosen printer. After that, the **Student** clicks the “Print” button and the **System** will check the **Student’s** remaining pages. If that is enough, the system will send the uploaded document after having specified by the **Student** to the selected printer and begin to print. Then, the **System** displays “Print successfully” to inform the **Student** and ends the “Print document” use-case. If the **Student** doesn’t have enough remaining pages for printing, the system will notify the **Student** and ask them if they want to buy more pages. If the **Student’s** answer is yes, the **System** will urge them to go to the BKPay website and buy more pages. After that, the **Student** can return to the “Printing” interface and click the “Print” button. If not, the **System** ends the “Print document” use-case. In the case that BKPay runs into an error and the **Student** can not buy any pages, the **System** will ask the **Student** to wait until the BKPay website operates normally and runs smoothly again. If the **Student** decides to wait, it will return to the BKPay website when the error is gone. Otherwise, end the “Print document” use-case.

2.4.5 Usecase: View student’s printing logs

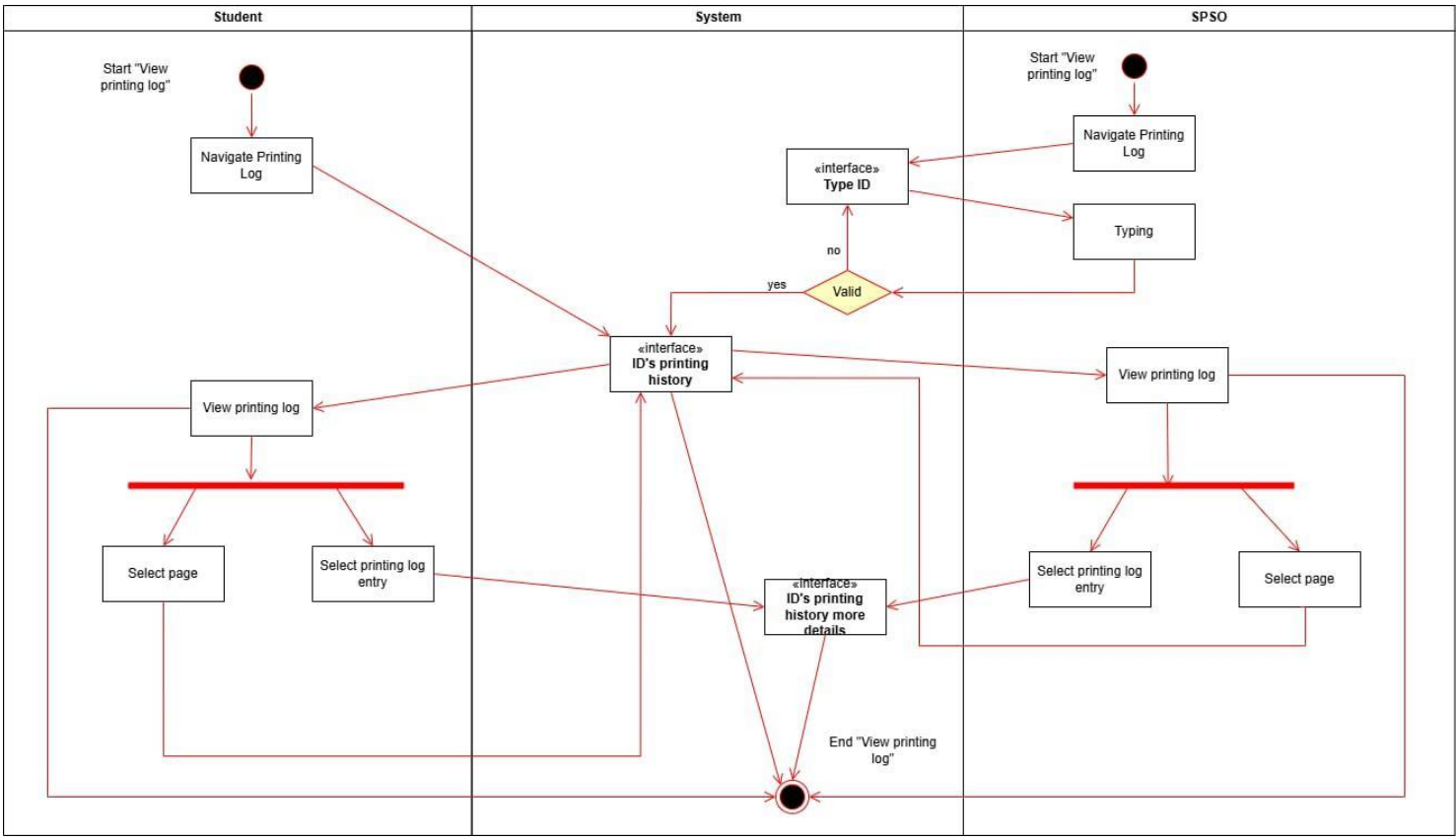


Image 7: The "View student's printing logs" Activity Diagram

The diagram outlines the process for the students and the SPSO to access the students' printing logs through a system interface. It is divided into three swimlanes, "**Student**", "**SPSO**" and "**System**", illustrating the interaction between the three. The process begins with the **Student** and the **SPSO** navigating to the "Printing log" section of the website. For the SPSO, after navigating, the **System** will show the "Type ID" interface and ask the **SPSO** to type in the **Student's** ID. If the ID is valid, the **System** will show the specified **Student's** printing history. Otherwise, return to the "Type ID" interface. After the "ID's printing history" interface is displayed, both the **Student** and the **SPSO** can view the printing logs and they can choose to select the specific page to find the log they want to see, select a log entry, or end the "View printing log" use-case. If they choose to select a specific page in the list of logs, the **System** will direct them to that page and show the "ID's printing history". If they select a log entry, the **System** will show more details with the "ID's printing history more details" interface and the **Student** or the **SPSO** can return to the "ID's printing history" to continue viewing the logs, repeating the process if necessary.

2.5 Sequence Diagram

2.5.1 Usecase: Upload document files

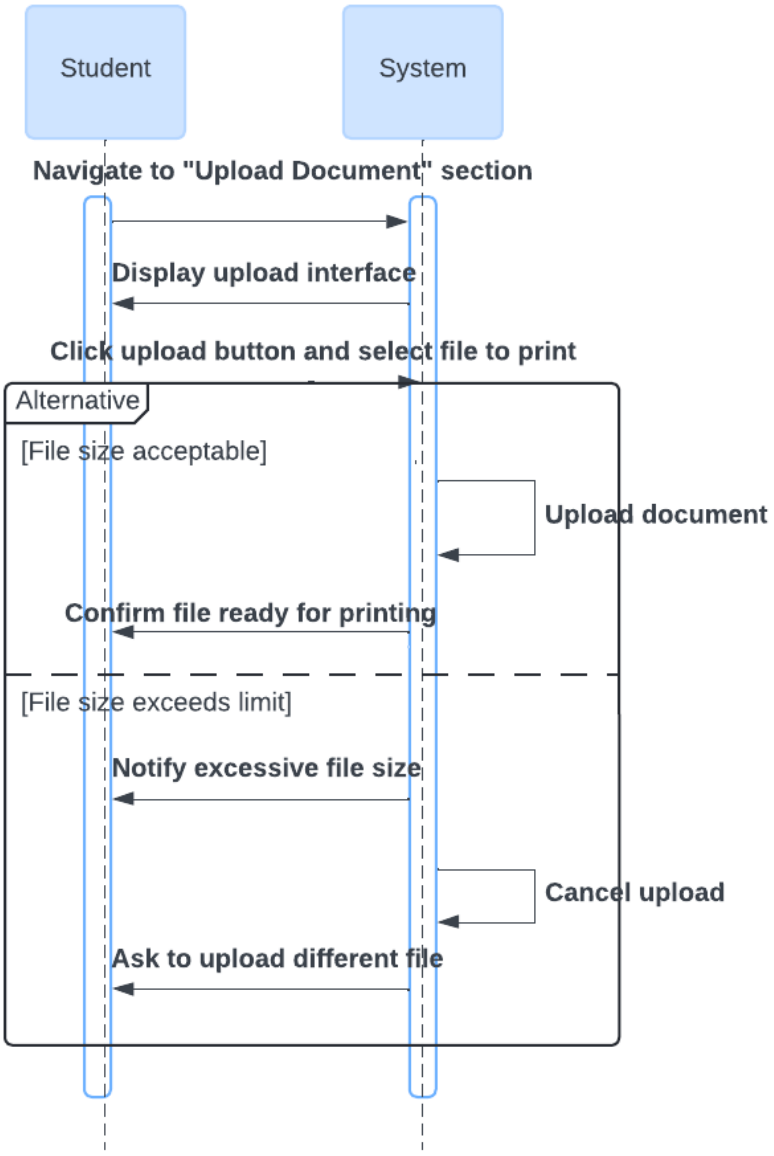


Image 8: The "Upload document files" Sequence Diagram

This sequence diagram illustrates the interaction between a **Student** and the **System** during a document upload process for printing. The **Student** first navigates to the "**Upload Document**" section, where the printing system displays the upload interface. The **Student** then selects a file for printing, and the **System** checks the file size. If the file

size is acceptable, the **System** uploads the document and confirms it is ready for printing. If the file size exceeds the limit, the **System** notifies the **Student** and cancels the upload, prompting them to upload a different file.

2.5.2 Usecase: Choose a printer

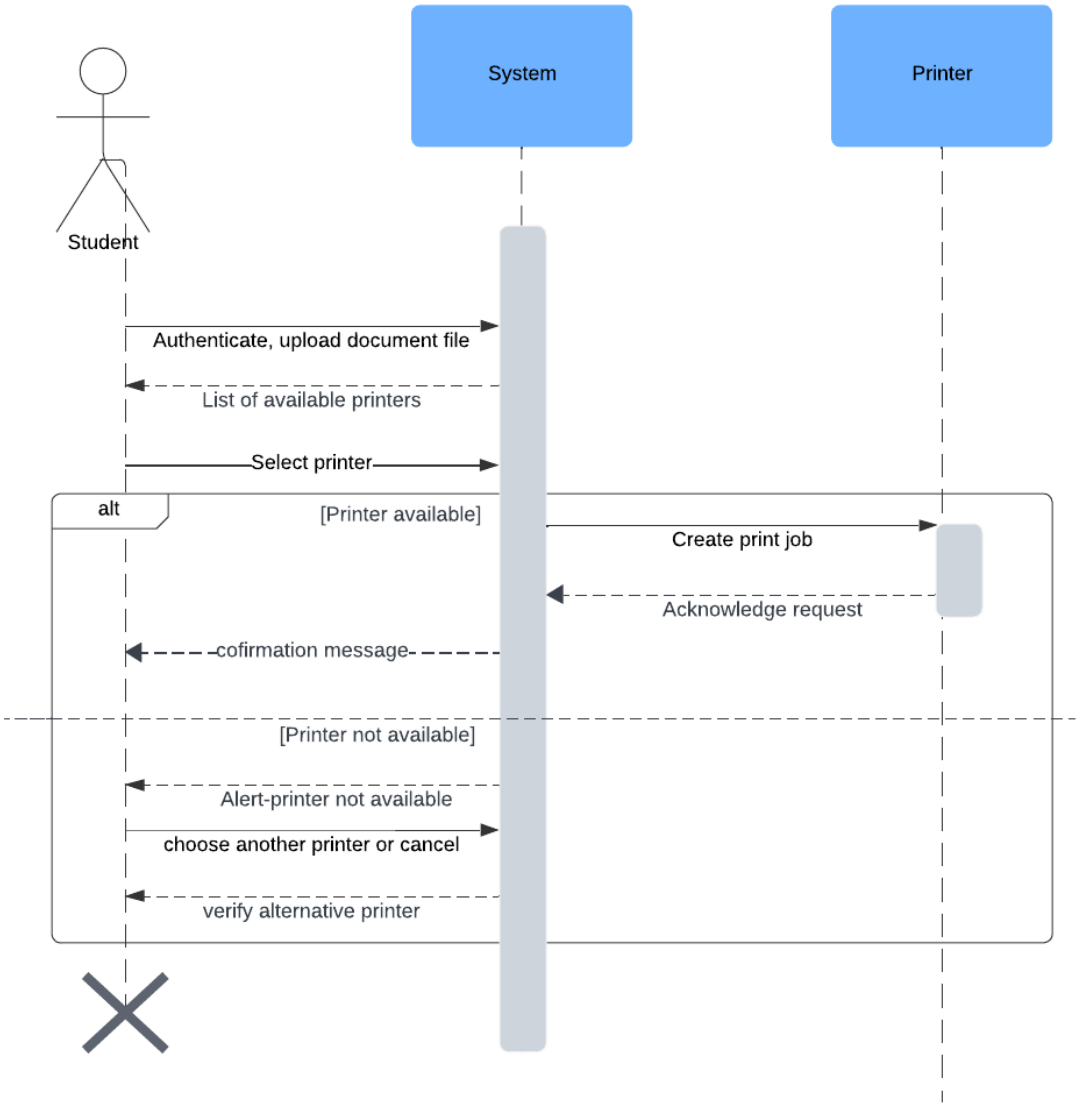


Image 9: The "Choose a printer" Sequence Diagram

This sequence diagram illustrates the "Choose a Printer" process for a student using the Student Smart Printing Service (SSPS) system. The process begins after the **Student** has authenticated and uploaded a document file to the **System**. In response, the **System** provides a list of available printers for selection. The **Student** then selects a

Printer, and the **System** checks its availability. If the printer is available, the **System** creates a print job and sends a confirmation message to the student, acknowledging the print request. If the **Printer** is not available, the **System** sends an alert notifying the **Student** of the unavailability. The **Student** is then given the option to either choose an alternative **Printer** or cancel the print request. If the **Student** selects an alternative **Printer**, the **System** verifies its availability, repeating the process as necessary.

2.5.3 Usecase: Specify Printing Properties

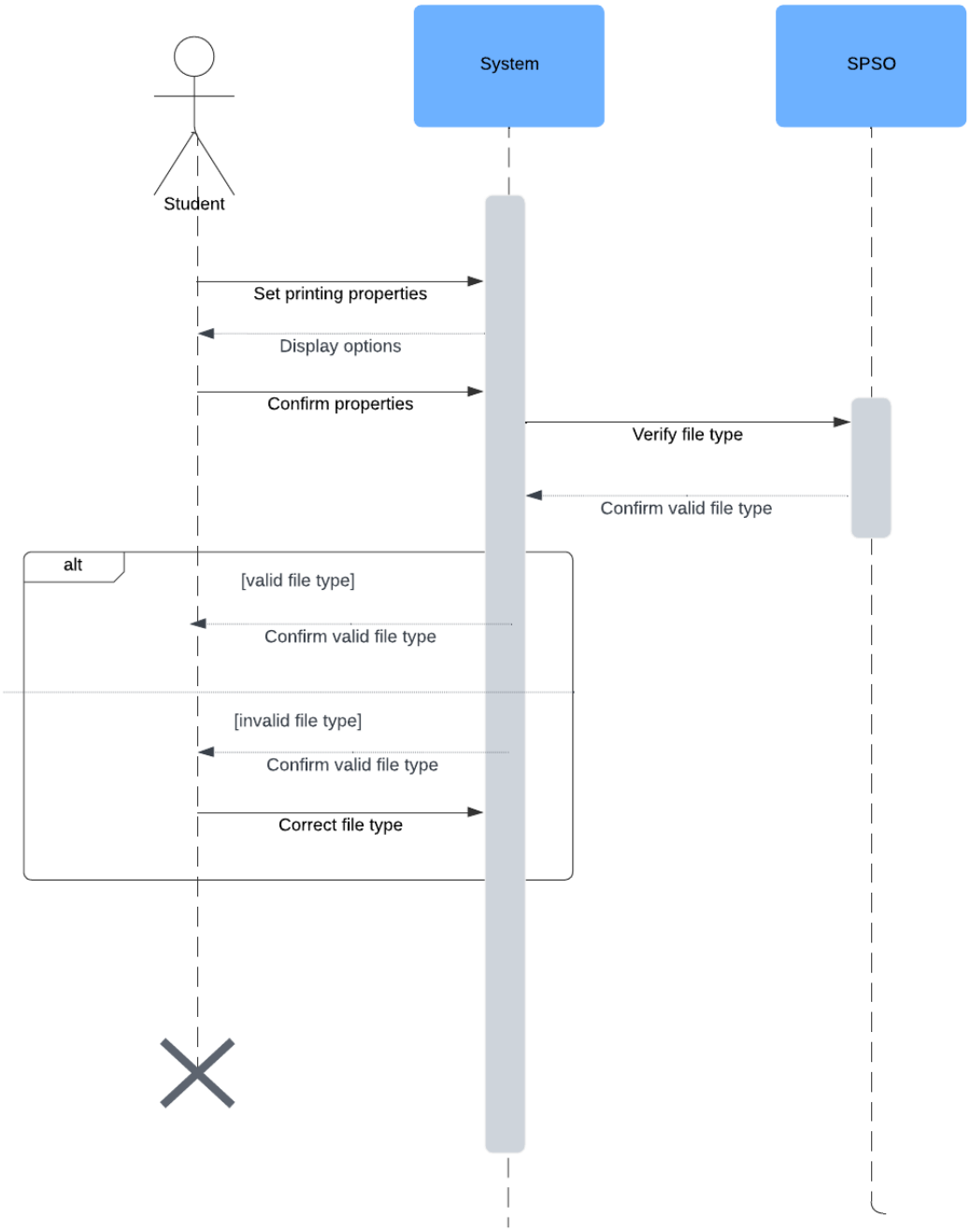


Image 10: The "Specify Printing Properties" Sequence Diagram

This sequence diagram illustrates the interaction between a **Student**, the **System**, and the **SPSO** during the process of setting and confirming printing properties. Initially, the **Student** sets the printing properties, which prompts the **System** to display the

available options. Upon reviewing the options, the **Student** confirms the selected properties. The System then verifies the file type by interacting with the **SPSO**. If the file type is valid, the **System** confirms the valid file type to the **Student**. However, if the file type is invalid, the **System** prompts the **Student** to upload another file with the correct type. This alternate flow ensures that only valid file types proceed to the printing stage, maintaining the integrity of the printing process.

2.5.4 Usecase: Print documents

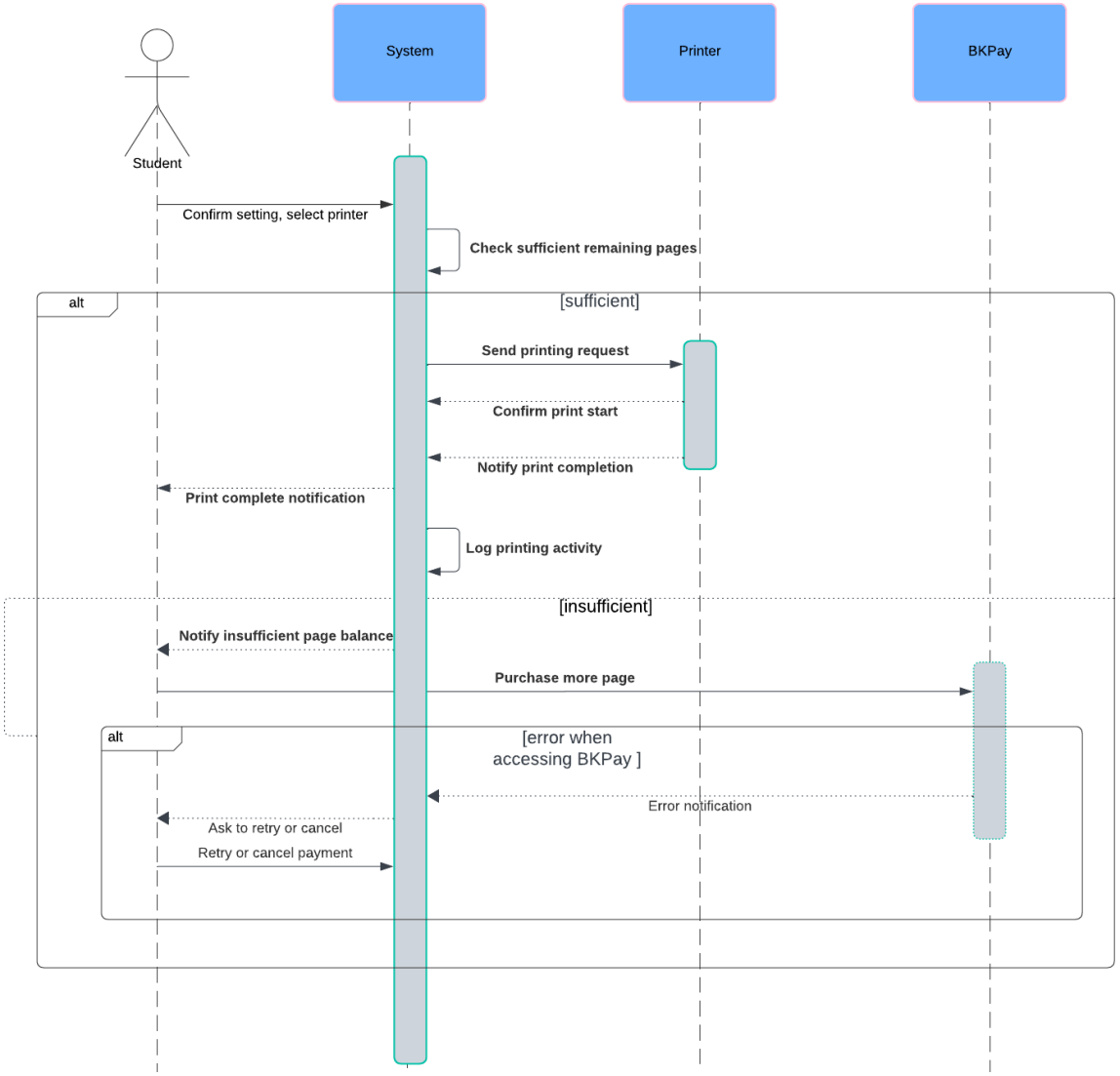


Image 11: The "Print documents" Sequence Diagram

This sequence diagram depicts the process a student follows to print documents, detailing the interactions between the **Student**, the **System**, the **Printer**, and **BKPay** (a

payment system). Initially, the **Student** confirms the settings and the selected **Printer**. The **System** checks for sufficient remaining pages. If the **Student** has enough pages, a printing request is sent to the **Printer**, which confirms the start of the print job and notifies the **System** upon completion. The **System** logs the printing activity and sends a notification of completion to the **Student**. Conversely, if the page balance is insufficient, the **Student** is informed and prompted to purchase more pages through the **BKPay**. In case of a payment error while accessing **BKPay**, the **Student** is given options to retry or cancel the transaction.

2.5.5 Usecase: View student's printing logs

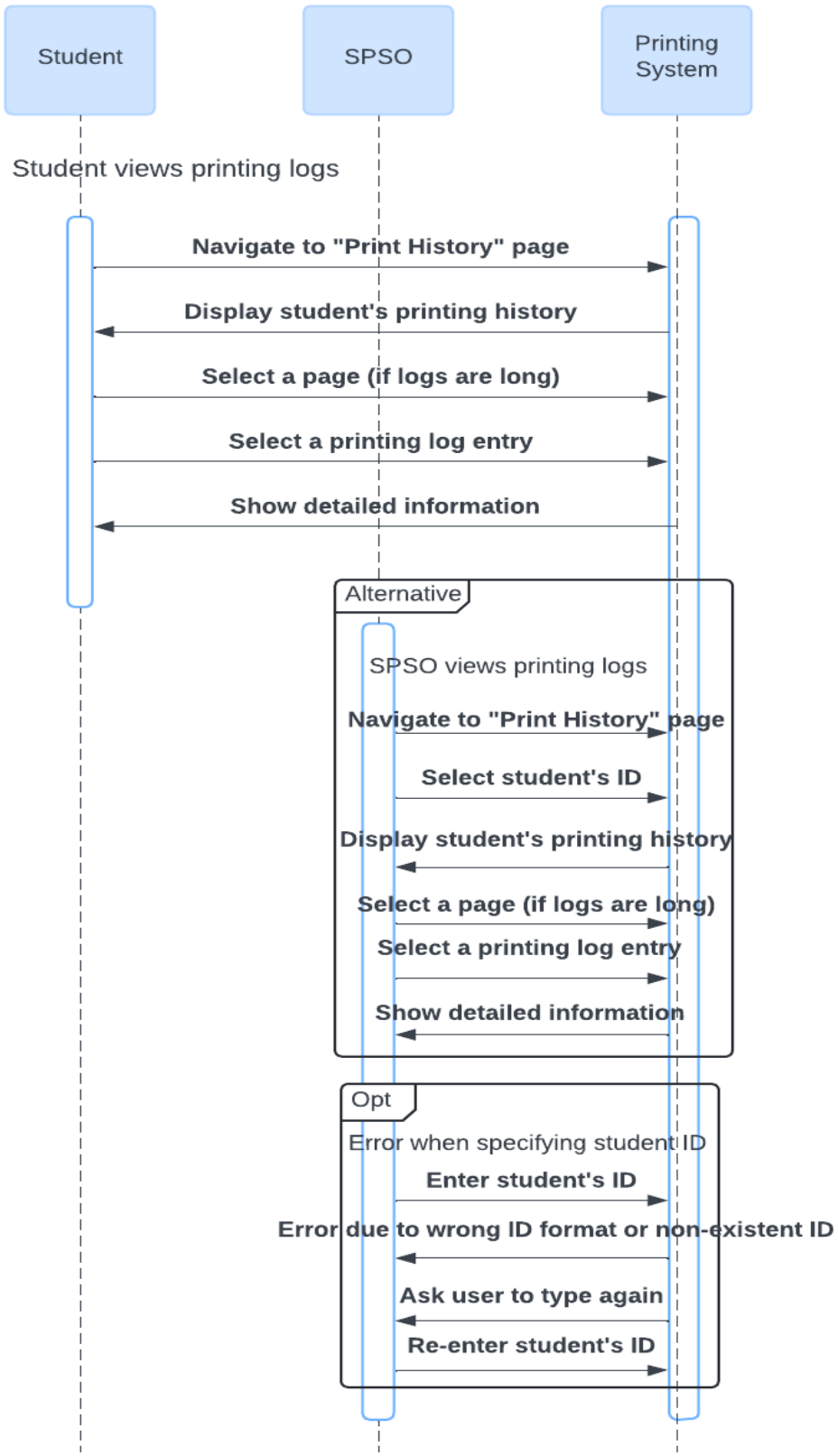

Student
SPSO
Printing System
Student views printing logs
Navigate to "Print History" page
Display student's printing history
Select a page (if logs are long)
Select a printing log entry
Show detailed information
Alternative
SPSO views printing logs
Navigate to "Print History" page
Select student's ID
Display student's printing history
Select a page (if logs are long)
Select a printing log entry
Show detailed information
Opt
Error when specifying student ID
Enter student's ID
Error due to wrong ID format or non-existent ID
Ask user to type again
Re-enter student's ID

Image 12: The "View student's printing logs" Sequence Diagram

This sequence diagram illustrates the process of viewing printing logs, both for a **Student** and a **Student Printing Service Officer (SPSO)**. The **Student** accesses the "**Print History**" page to view their own printing history. If the logs are lengthy, the **Student** can select a page and then choose a specific printing log entry to see detailed information. Alternatively, the **SPSO** can view a **Student's** printing logs by first navigating to the "**Print History**" page and selecting the **Student's** ID. The **System** then displays the **Student's** printing history, where the **SPSO** can also page through logs and select an entry for details. An optional scenario handles errors when the **SPSO** enters an invalid or incorrectly formatted **Student's** ID, prompting the user to re-enter the ID.

2.6 Class Diagram

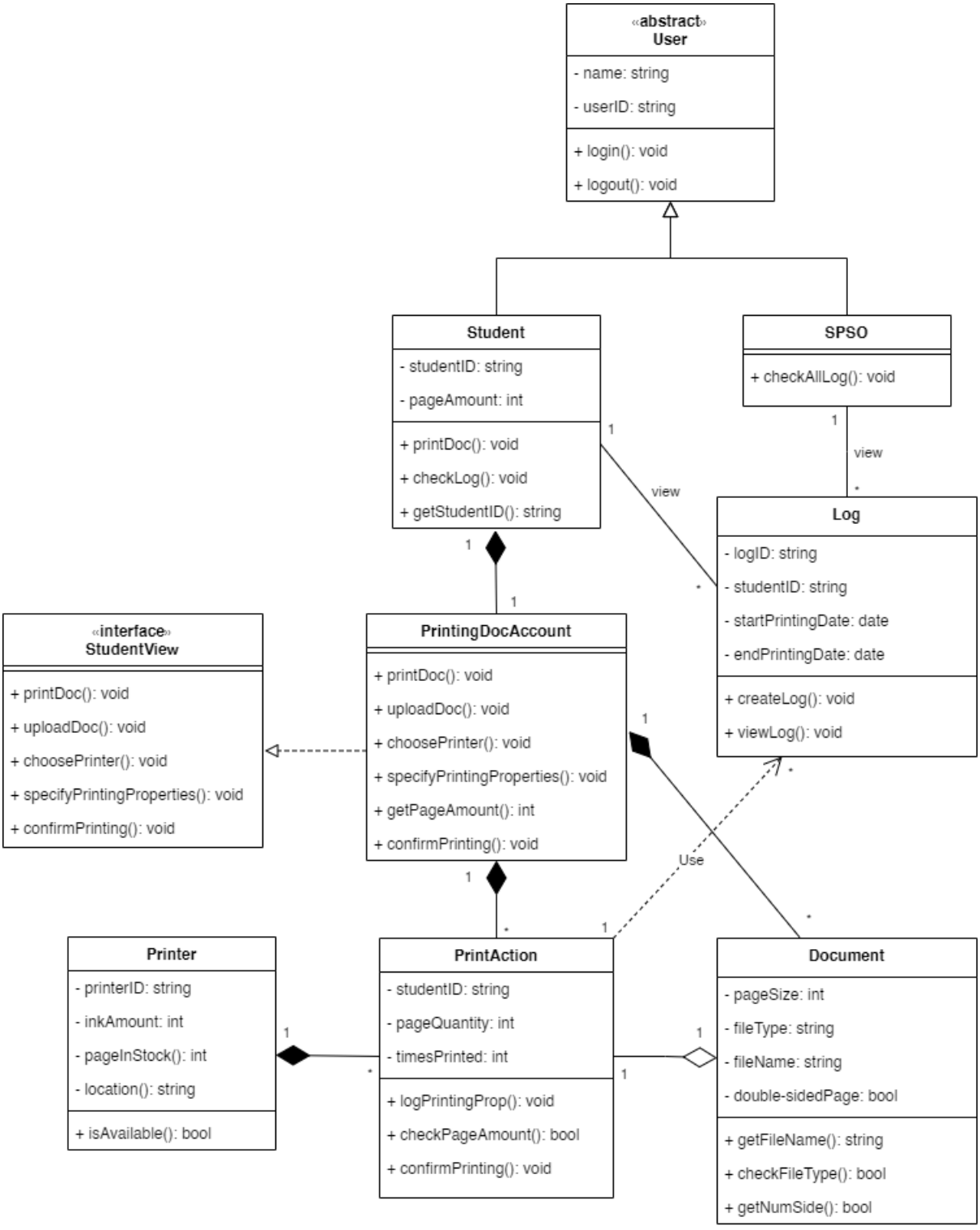


Image 13: The "Print documents" module Class Diagram

This class diagram represents the “Print documents” module in a Student Smart Printing Service (HCMUT_SSPS) system, with key classes and their relationships. The module is centered around the **Student** class, which inherits from the abstract **User** class that defines common properties and methods for all users, such as login() and logout(). Each student has a **PrintingDocAccount** that allows them to print documents, upload documents, choose printers, specify printing properties, and check their remaining page amount. The **StudentView** interface provides methods that allow students to interact with the account, including printing, uploading, and confirming print jobs. The **SPSO** class is responsible for managing logs through methods like checkAllLog(), allowing the printing service operator to monitor printing activities. The **Log** class records each print transaction, containing details such as logID, studentID, startPrintingDate, and endPrintingDate. Each **PrintAction** is associated with a student and tracks information such as page quantity and times printed. The **Printer** class represents individual printers in the system, containing attributes like printerID, inkAmount, pageInStock, and location. Each printer can be checked by the system for its availability through the isAvailable() method. Finally, the **Document** class represents the files students wish to print, storing details such as pageSize, fileType, fileName, and whether it is double-sided. Methods like getFileName(), checkFileType(), and getNumSide() allow document property checks to ensure compatibility with printing requirements.

2.7 User Interface

Based on the above diagrams, the UI interface was designed to visualize the app and this section only focuses on the “Print documents” module. For more interactive means, please refer to this Figma design.

- Figma Prototype: MVP UI

Together this module’s design consists of 10 pages which show the interface of the HCMUT_SPSS website.

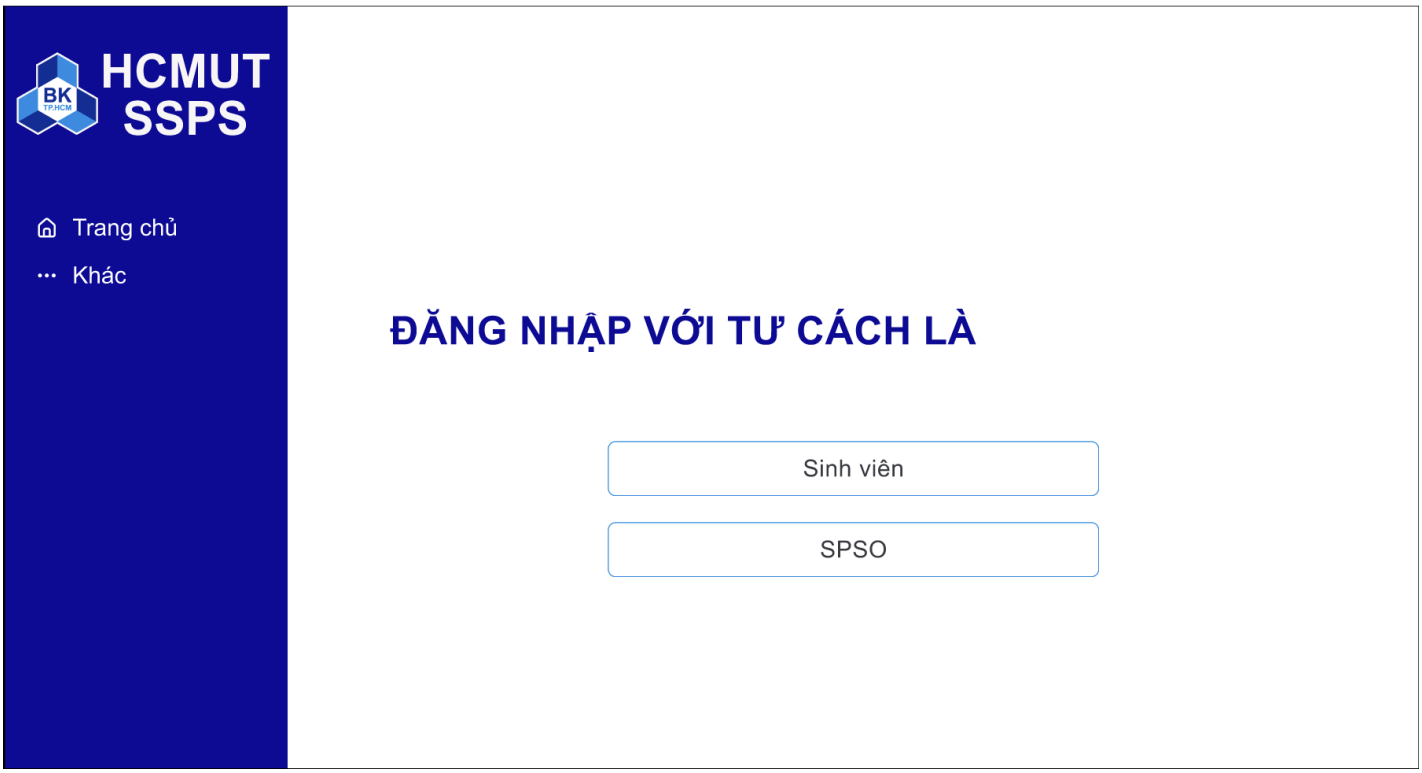


Image 13: The User has to choose to login whether as a student or as a SPSO

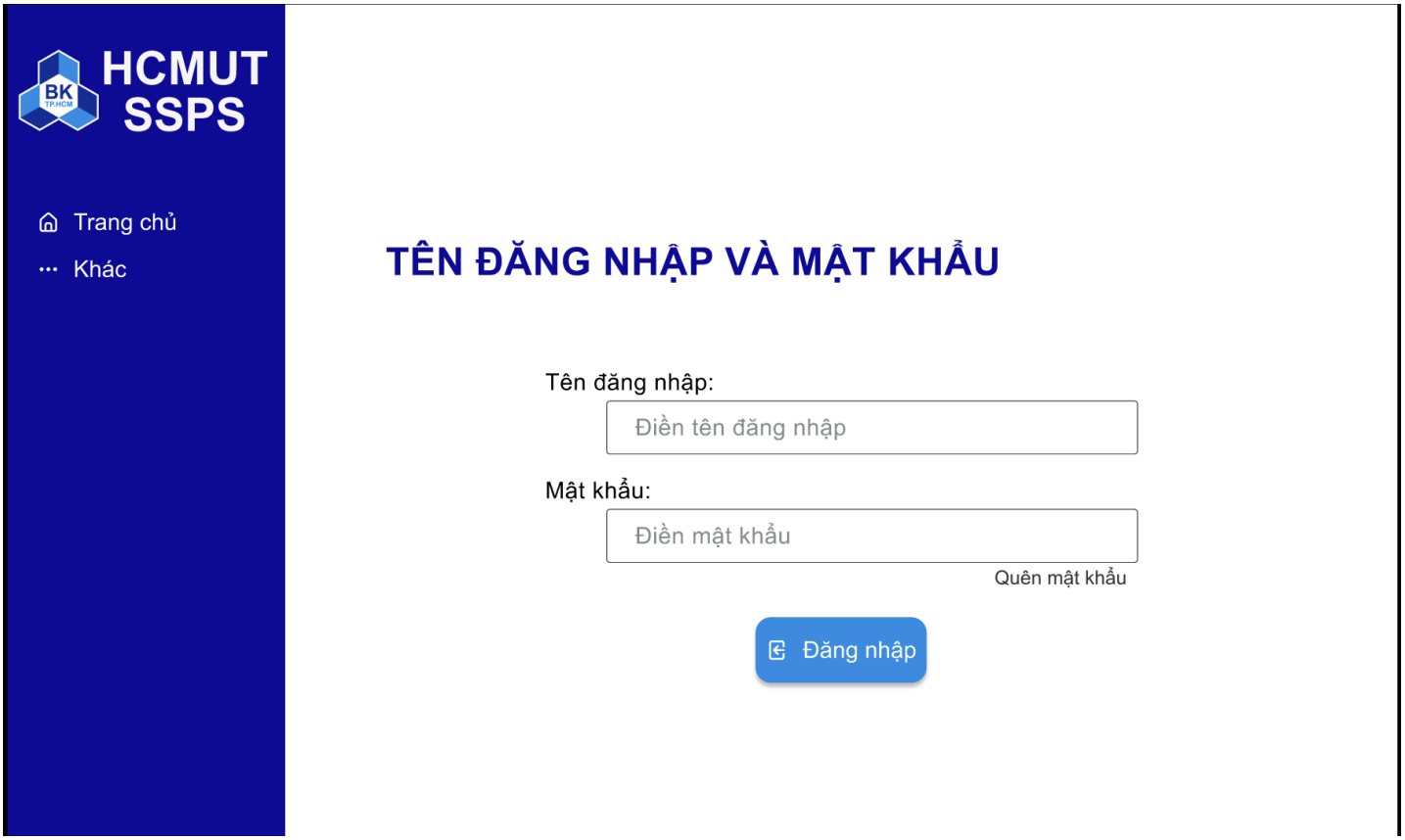


Image 14: The User's login page

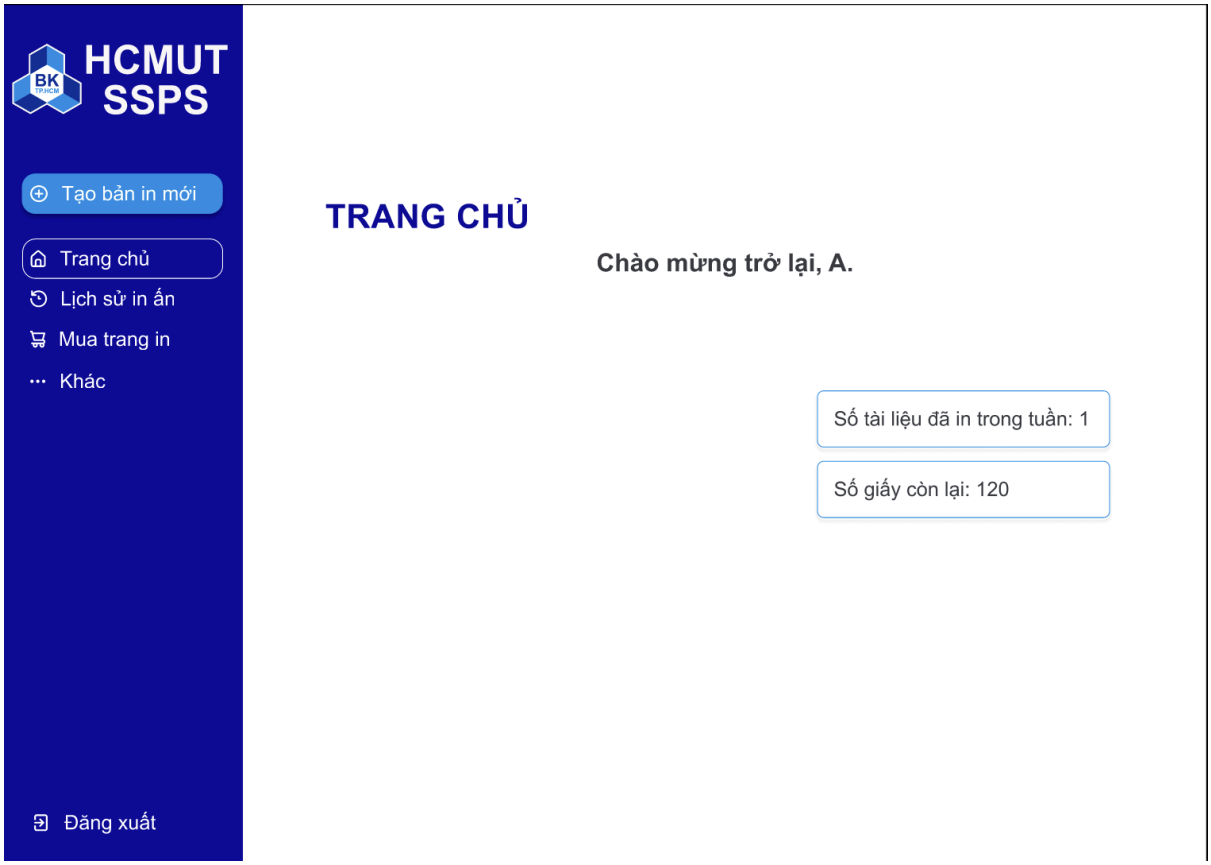


Image 15: The Student's homepage after login

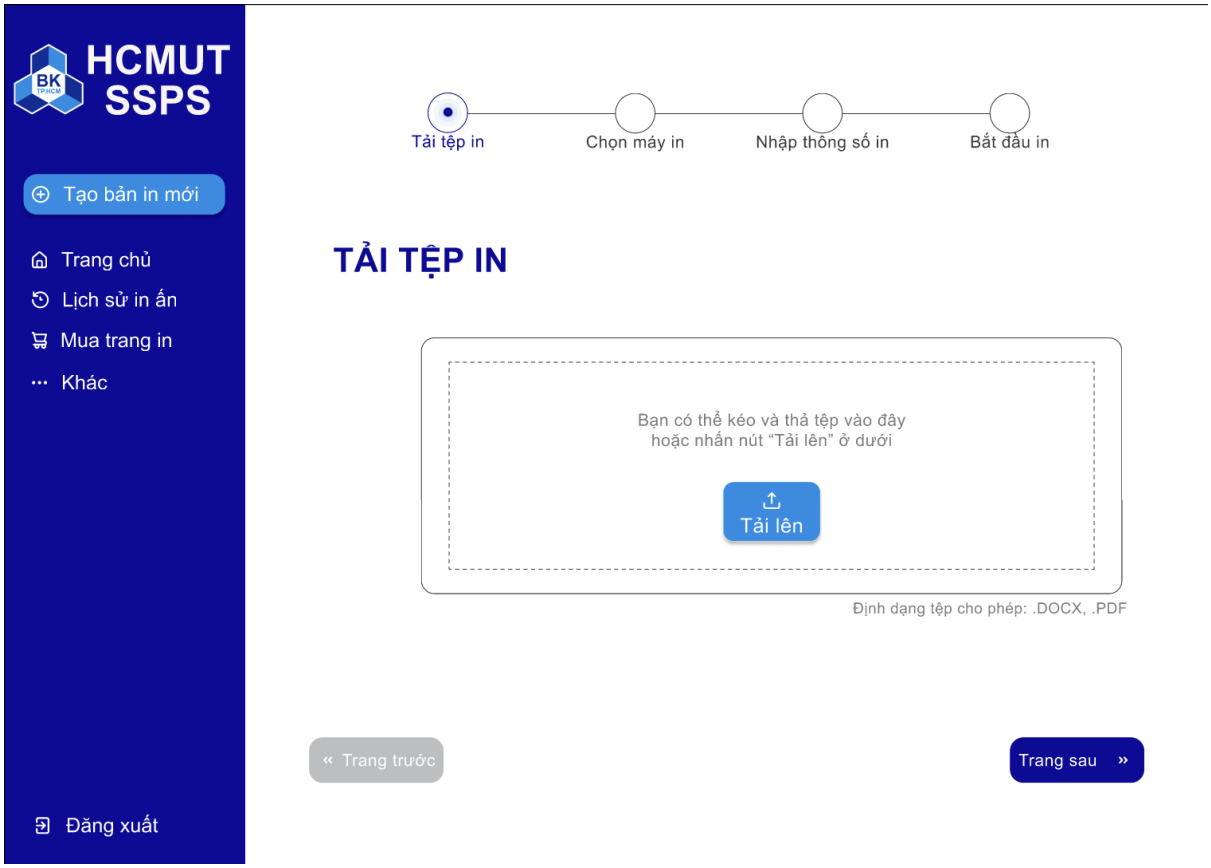


Image 16: The first step in the process of printing documents after clicking the "New printing job" button, the page where the Student can upload their files

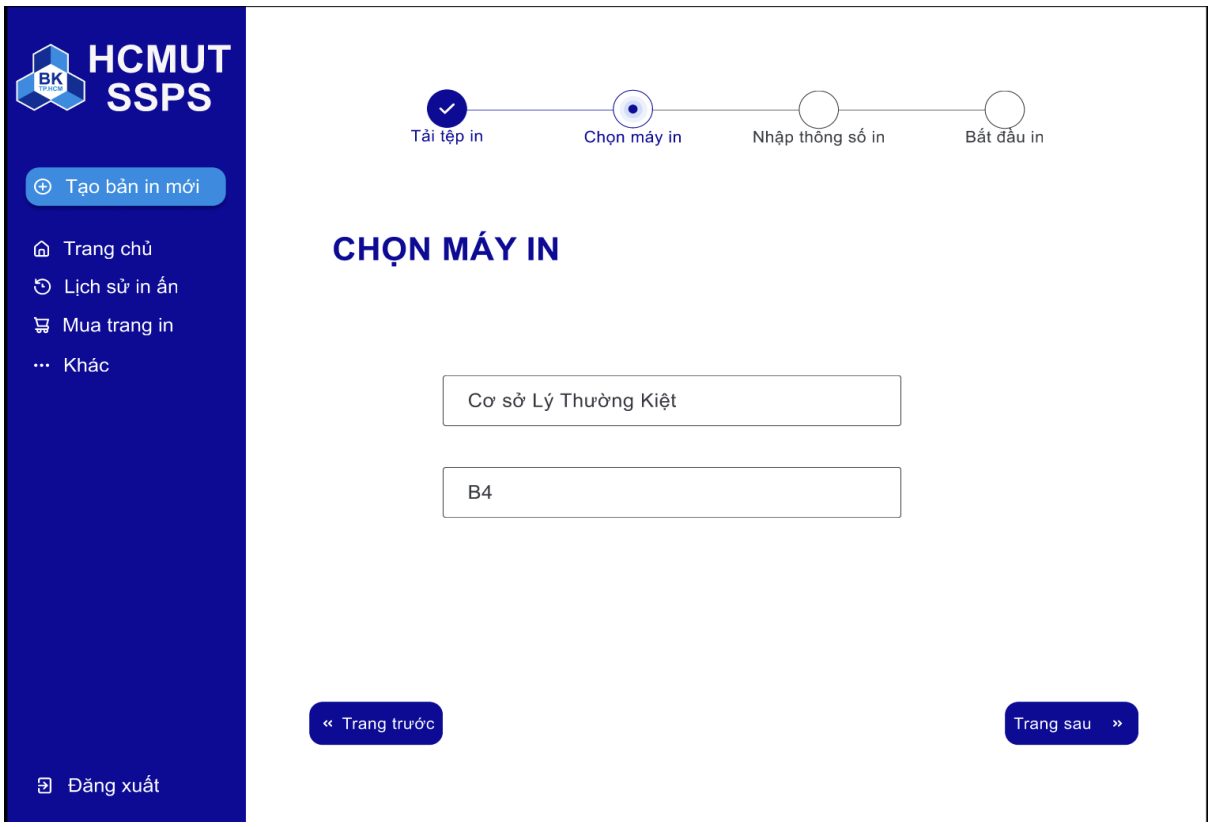


Image 17: The second step in the process of printing documents, the page where the Student can choose the printer's location.

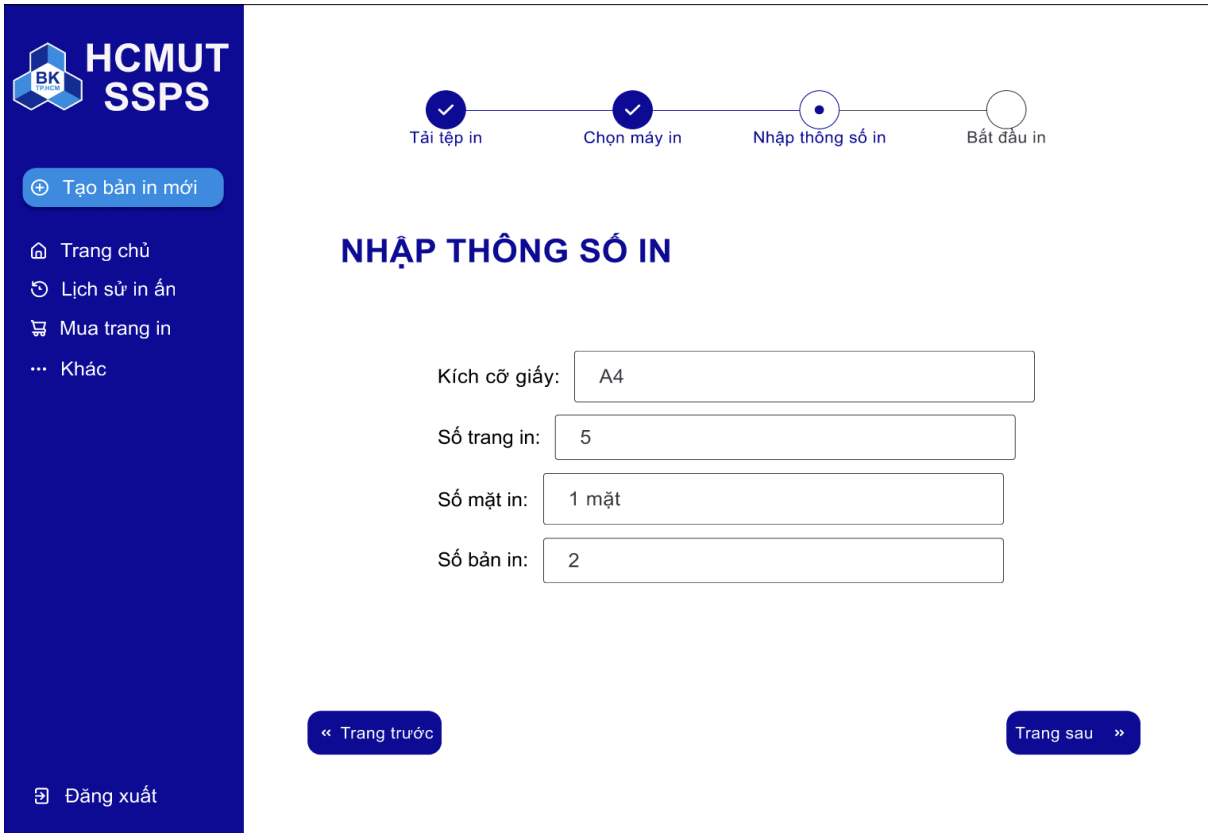


Image 18: The third step in the process of printing documents, the page where the Student can specify the printing properties.

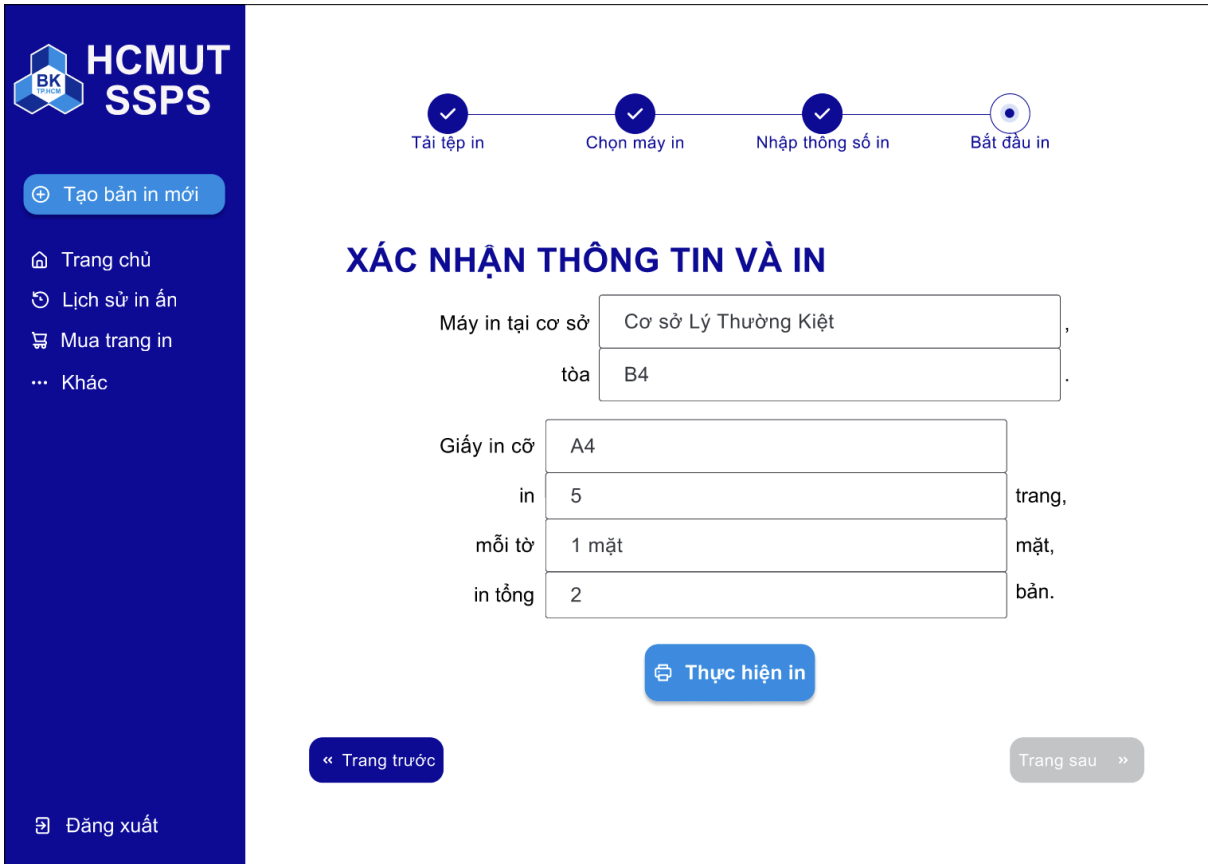


Image 18: The preview page of the settings and the "Print" button.

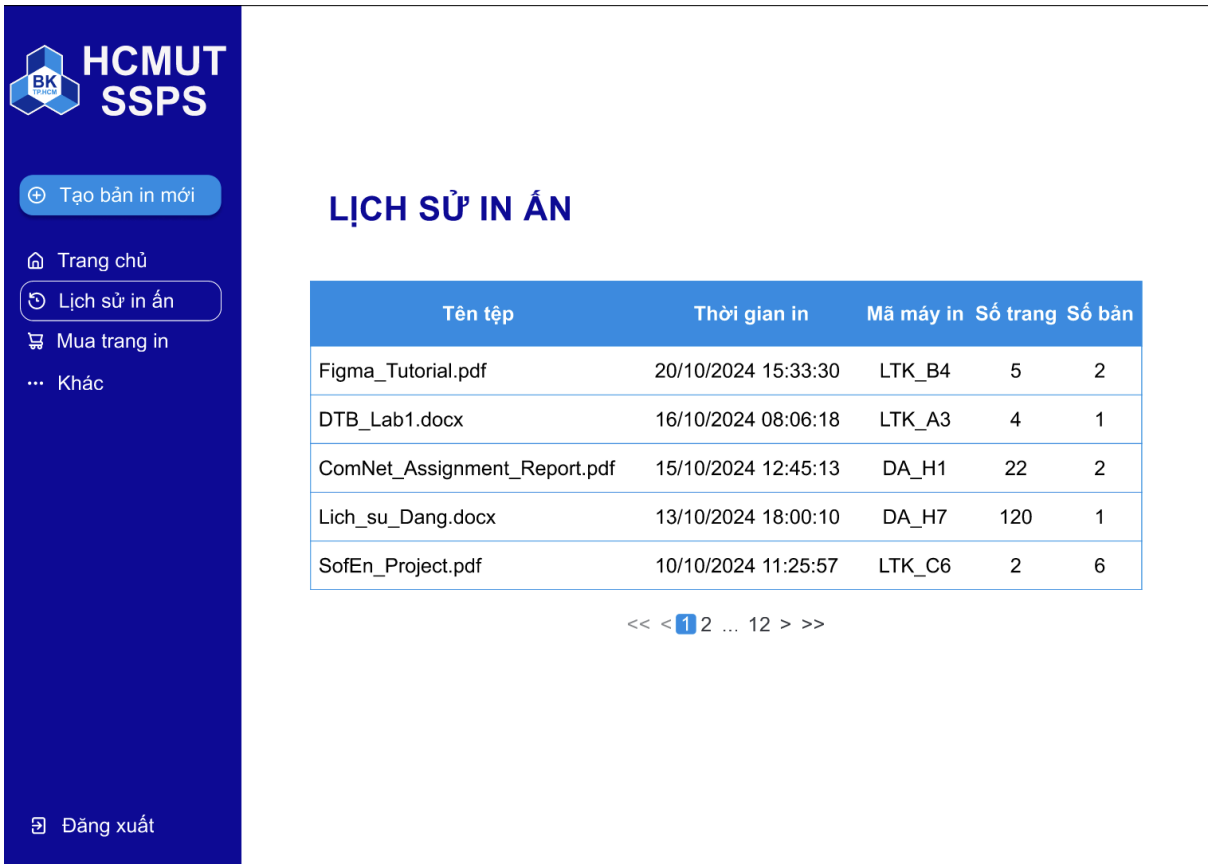


Image 19: The printing logs of the Student.

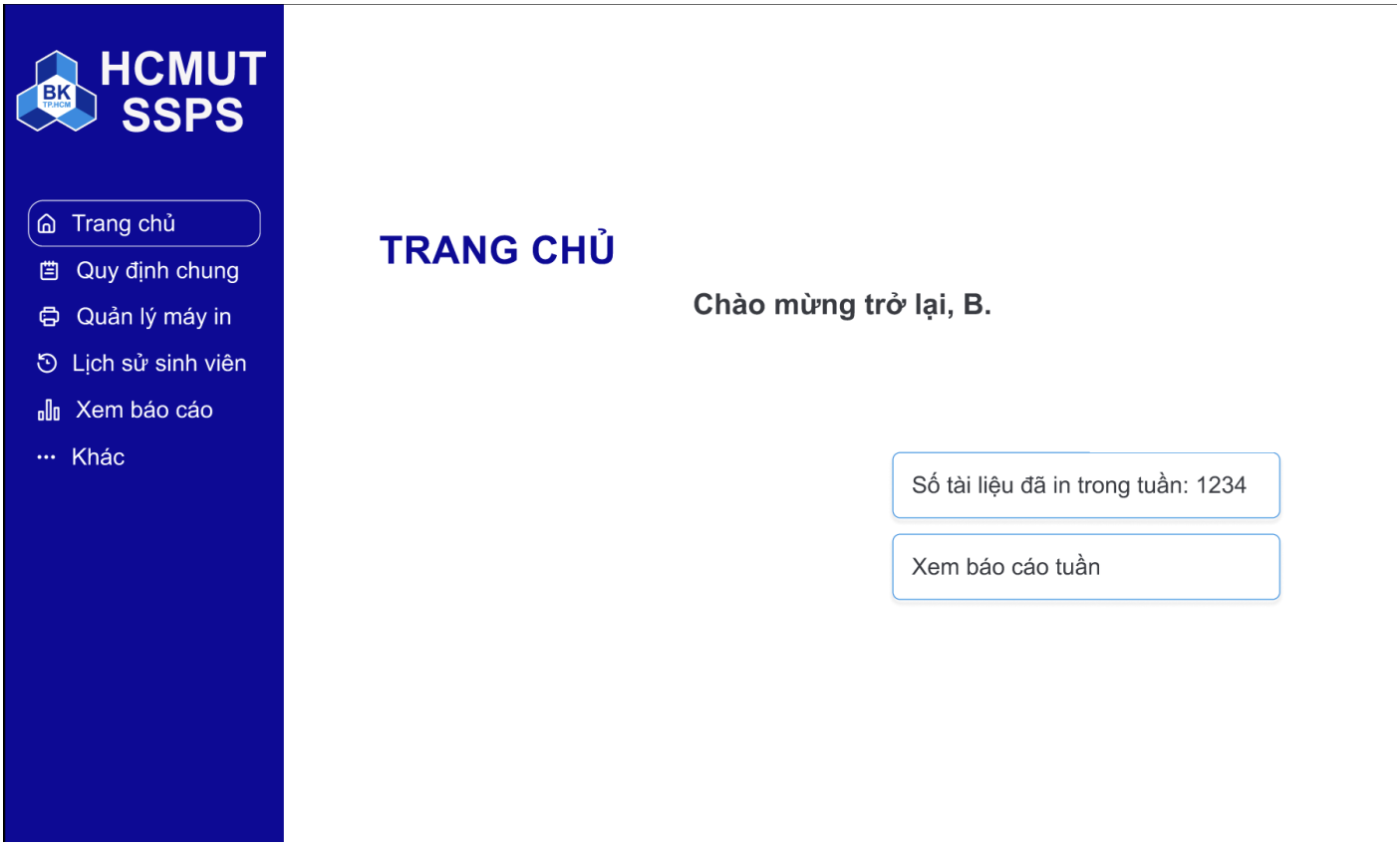


Image 20: The SPSO's homepage after login

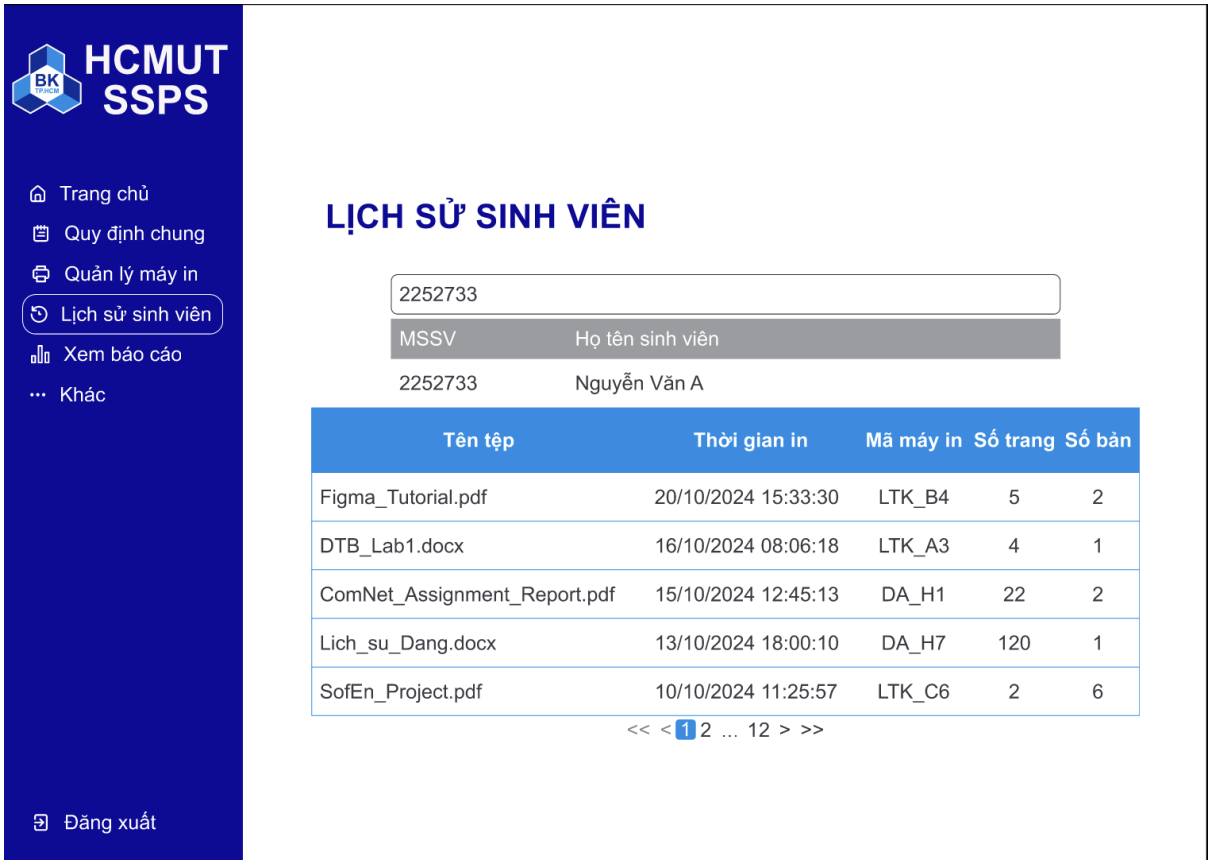


Image 21: The SPSO can view any of the Student's logs by specifying the Student's ID

3. Architecture design

3.1 Layered architecture

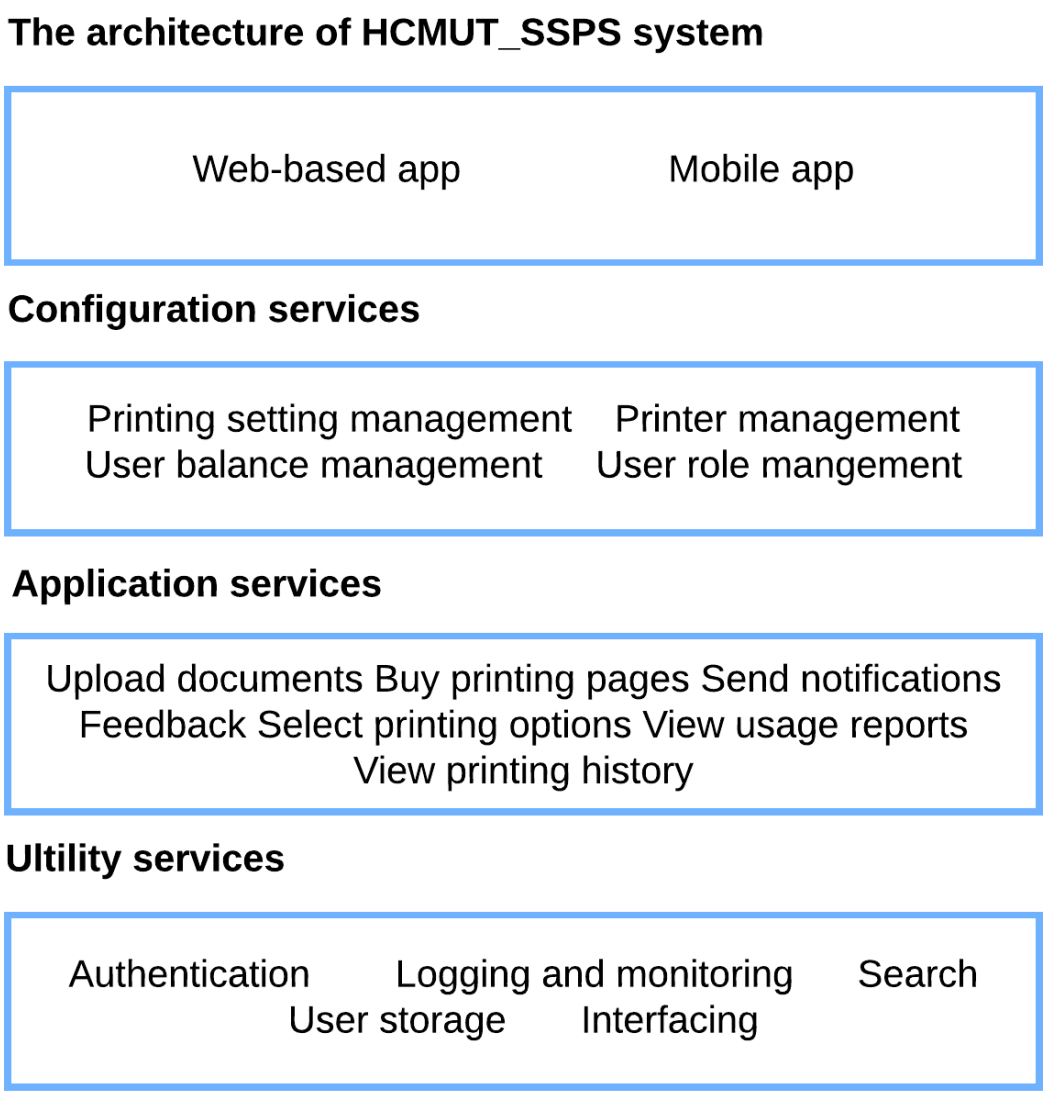


Image 22: Architectural diagram of the HCMUT-SSPS system.

3.1.1 Presentation strategy

The UI design strategy for the HCMUT Smart Printing Service shall focus on delivering a user-friendly, responsive, and role-based user interface with the aim of smoothing

printing workflows and serving various needs of students, SPSOs, IT support, and university administrators. It ensures intuitive navigation wherein users must be able to upload documents, set properties for printing, manage printers, or generate reports with ease. Clarity to the user is provided through visual cues like status indicators, error notifications, and step-by-step guidance. These will reduce the learning curve and minimize errors within the system. Automatically adjust to a wide range of devices and screen sizes for user access from campus or remote locations. The system features role-based dashboards that customize the display to only present those features most germane to each profile of user type. This provides the most efficient means of navigation and reduces cognitive load. It integrates real-time feedback, clear confirmation dialogs, and dynamic data related to printer availability and usage logs, reinforcing transparency and thus building the user's trust. The approach enables easy operation of the system and maintains a positive and engaging user experience.

3.1.2 Data storage approach

The data storage approach involves systematically organizing and managing data to ensure seamless functionality and easy access. User data, such as profile information and balances, is stored in a relational database like MySQL or PostgreSQL, which allows for structured queries and data integrity. Uploaded documents are stored in cloud storage solutions such as AWS S3 or Google Cloud Storage, ensuring scalability and reliable access. Printing properties, including settings like paper size and print quality, are stored in a NoSQL database like MongoDB for flexible and scalable management. Printer resource information, including printer availability and location, is also maintained in a relational database. Logs and error tracking are handled by specialized logging services such as ELK Stack or AWS CloudWatch, which provide real-time monitoring and troubleshooting capabilities. Authentication data is managed by dedicated services like Auth0 or Firebase Authentication to ensure secure user sessions. This multi-faceted approach to data storage ensures that all components of the system are efficiently managed, secure, and easily accessible, facilitating smooth and reliable operations for the HCMUT_SSPS system.

3.1.3 API management

To manage the APIs for the HCMUT_SSPS, we need an effective API management strategy considering some aspects: gateway configuration, developer portal, API lifecycle.

a) API gateway

The API Gateway would serve as the central point of access for all API requests to the HCMUT_SSPS, including document uploads, printing management, logging, and payment processing, including:

- Directing API requests to the appropriate service endpoints, balancing the load across servers to handle the high concurrency needs
- Enforcing role-based access control for students, SPSOs, IT support, and administration, and ensuring authentication through HCMUT's SSO integration
- Preventing excessive usage by limiting requests, particularly during peak usage times (e.g., exam periods) to prevent degradation of service.

We use Axure API

b) Developer Portal

The Developer Portal would be the central hub for us to access API documentation, developer resources. The portal should provide clearly documented endpoints for functionalities like document upload, printer selection, print settings configuration, payment processing, and log retrieval. Besides, the portal should also provide a sandbox environment, letting developers simulate API calls with mock data, facilitating a safe area for building and testing integrations without affecting live data.

c) API lifecycle management

This part is crucial to maintain stability, flexibility, and functionality as the system evolves. We need to collaborate with stakeholders to determine functional requirements (e.g., upload document, log printing activities) and design APIs accordingly. Each major update would also increment the API version (e.g., v1, v2), allowing older versions to coexist for backward compatibility

Considering all the aspects, we use Azure API Management for designing and monitoring APIs for this project.

https://learn.microsoft.com/en-us/azure/api-management/api-management-key-concepts
https://www.youtube.com/watch?v=fh3VaXLzH5Y&t=372s

3.2 Component diagram

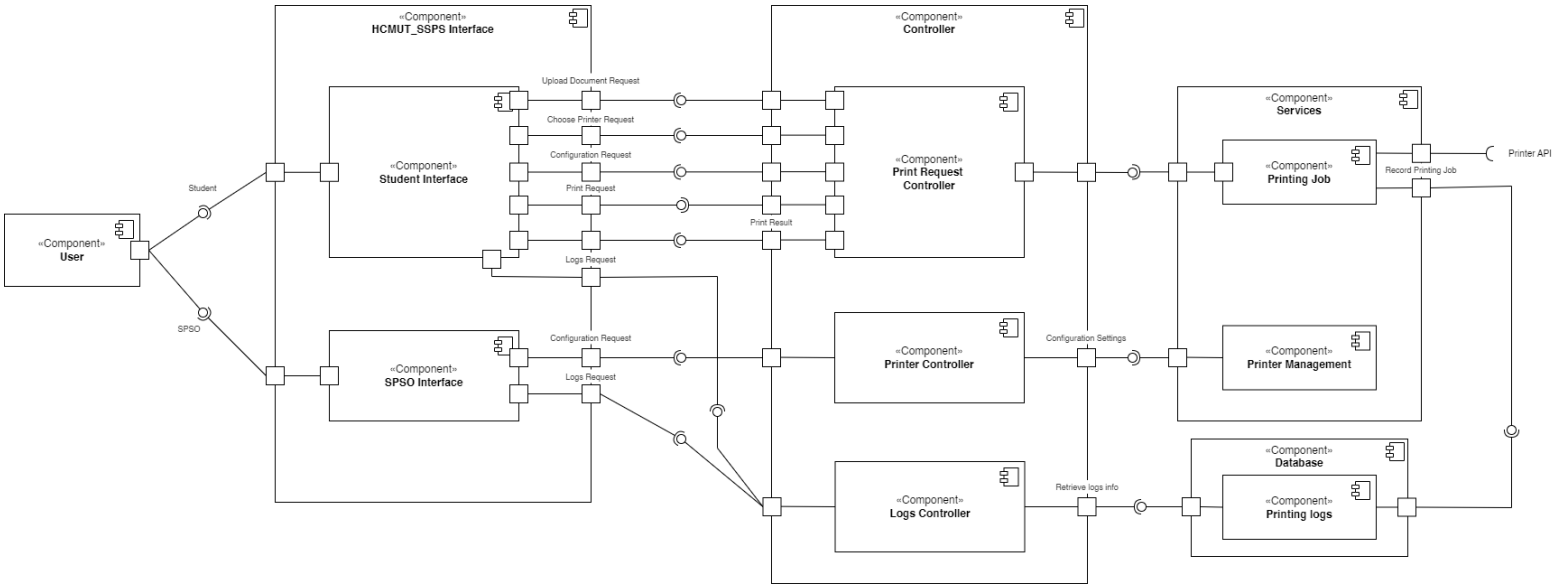


Image 23: Component diagram of the "Print documents" module

The component diagram for the HCMUT Student Smart Printing Service (SSPS) showcases a sophisticated and structured approach to managing student printing needs across the university's campuses. At the core of this system are various interconnected components designed to ensure seamless functionality and user convenience. The users, comprising students and the Student Printing Service Officer (SPSO), access the system through the HCMUT_SSPS Interface, which bifurcates into the Student Interface and the SPSO Interface. Students interact with the Student Interface to upload documents, choose printers, configure printing properties, and view their printing logs and balances. Meanwhile, the SPSO uses the SPSO Interface to manage printer settings, view comprehensive logs, adjust system configurations, and generate necessary reports. The Controller section, which includes the Print Request Controller, Printer Controller, and Logs Controller, plays a pivotal role in handling the print requests, managing printer operations, and maintaining detailed logs of all printing activities. The Print Request Controller processes requests related to document uploads, printer selections, and configurations, while the Printer Controller ensures the efficient execution of print jobs and manages printer statuses. The Logs Controller is responsible for storing and retrieving detailed logs, which include information such as student ID, printer ID, file names, and the specifics of the printing job. The Services component, which includes the Printing Job and Printer Management, directly oversees the printing tasks and interacts with printer hardware via the Printer API, ensuring accurate execution of print jobs.

Printer Management also deals with configuration settings and log management. The Database component, specifically the Printing Logs, stores all detailed records of print jobs for future reference and reporting. This system is built to ensure that students can easily manage their printing needs, with features allowing them to purchase additional printing pages through integrated online payment systems like BKPay, ensuring they stay within their allocated balance. The system also enforces authentication through the HCMUT_SSO service, guaranteeing secure access for all users. Automated report generation at monthly and yearly intervals provides the SPSO with vital insights into printing usage and system performance. This well-designed architecture, accessible via both web-based and mobile applications, ensures that the HCMUT Student Smart Printing Service operates efficiently, meeting the diverse needs of the student body while maintaining robust control and oversight through the SPSO. This comprehensive system underscores the university's commitment to leveraging technology to enhance student services, ensuring that printing tasks are managed effectively and transparently across the campus.

4. Implementation Sprint 1.

4.1 Setting up an online repository with Github.

Link: https://github.com/tamnd04/Project-SE-CC04-06

- Readme file:

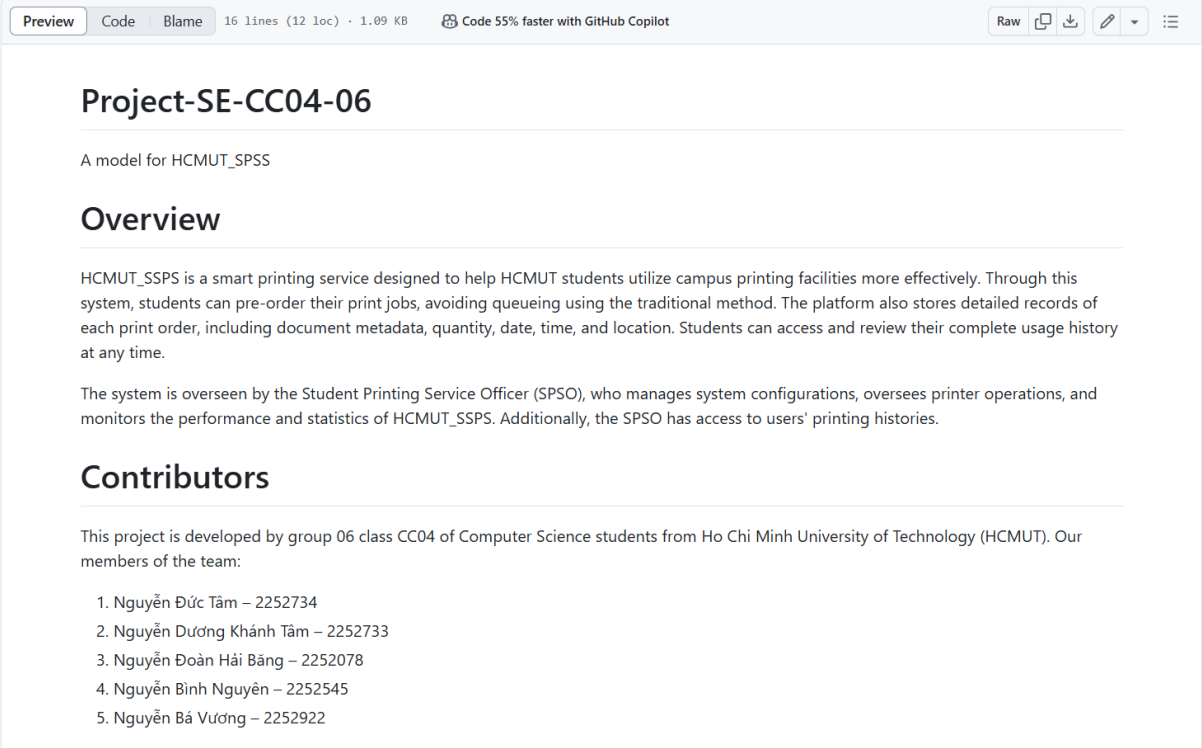


Image 24: The Readme file

4.2 Conducting usability test with the MVP 1 interface.

5.2.1. Recruit participants/ testers.

List of participants/testers in user interface testing:
• Nguyen Duc Tam
• Nguyen Duong Khanh Tam
• Nguyen Doan Hai Bang
• Nguyen Binh Nguyen
• Nguyen Ba Vuong
The participants/tester will play the role of students/admin using SSPS software to perform the printing task.

5.2.2. Define task.

There are in total 8 tasks that the users need to perform in order to complete the usability tests. The tasks include:
a) Home page

- Go to the login page for student by clicking the button “Sinh viên”

- Go to the login page for admin by clicking the button “SPSO”

b) *The User’s login page*

- Login into the system as a student, using the following information: Username: studentID. Password: yourpassword.
- Login into the system as an admin, the username and password is provided when developed the system.

c) The user homepage after login

- Turn to printing documents by clicking the “New printing job” button.

d) Upload documents page

- Upload the acceptable document file(.docx, .pdf): “Browse file from your computer” or “Drag and drop file”
- Delete the uploaded document file.
- Reupload the document file.
- Go to next page.

e) Select printer page

- Choose the printer in specific location and building.
- Rechoose the printer.
- Go to the previous page.
- Go to the next page.

f) Printing properties pages.

- Specify the printing properties.
- Respecify the printing properties.
- Go to the previous page.
- Go to the next page.

g) The preview page.

- Review the printing document.
- Go to the previous page

- Start printing by clicking “Print”

h) Printing log page

The students’ printing log page:

- View the printing log.
- Choose the direct page for specific printing log.
- Choose the printing log for more details information.

The SPSO’ printing log page:

- View any of the Student’s logs by specifying the Student’s ID
- View the printing log.
- Choose the direct page for specific printing log.
- Choose the printing log for more details information.

5.2.3. Define Test Strategy

A test strategy is the direction the testers want to evaluate through various testing methods.

Testing Direction: There are two directions for usability testing: qualitative and quantitative.
- Qualitative Testing: Focuses on the thoughts, feelings, and actions of participants when they use the product. Qualitative testing is best suited for identifying issues in the user experience.
- Quantitative Testing: Focuses on quantifiable measures of the user experience, including the number of tasks completed and the time taken to complete each task.

Testing Methods: The evaluation can be conducted in two ways: in-person or remote. Remote evaluation is further divided into "Moderated" and "Unmoderated."
- Moderated Method: Similar to in-person methods, both require direct communication between the developer and the participant. However, moderated remote testing differs from in-person testing in that the developer and participant are in different geographic locations, requiring software such as Google Meet, Zoom, etc., to communicate.
- Unmoderated Method: Uses application software that supports task creation, where the software acts as the developer, providing, guiding, and assisting participants in completing tasks.

Based on the theoretical basis of these methods, we chose the unmoderated remote testing method to evaluate the product quality due to its speed and convenience for participants. It eliminates geographical barriers and allows for flexibility in testing times. We also chose qualitative testing to comprehensively understand the user experience while partially applying quantitative methods by tracking the tasks performed and completed.

5.2.4. Conduct the test.

Test overview: We conduct the test of HCMUT_SPSS to evaluate the functionality, usability, and overall user experience of the application for both student/ admin users. The primary focus was on identifying usability challenges and areas for improvement across eight key tasks defined in the application's user workflow.

Test Details

- **Application Tested:** HCMUT_SPSS (Student Printing Support System)
- **Test Methodology:**
  - Moderated usability test, conducted remotely.
  - Tasks were completed in sequence while participants were observed through a streaming website (e.g. Google Meet).
  - Data collected included success rates, errors, time on tasks, and participant feedback.
- **Environment:**
  - Devices: Laptop with the latest web browser installed.
  - Controlled testing environment with minimal distractions.
  - Standard internet connection speed (5 Mbps).
- **Duration:**
  - Each session lasted approximately **15 minutes**, with time allocated for task completion and brief participant feedback

Task	**Nguyen Duc Tam**	**Nguyen Duong Khanh Tam**	**Nguyen Doan Hai Bang**	**Nguyen Binh Nguyen**	**Nguyen Ba Vuong**
1. Home Page	Success: 100% Time: 1 min Error: Misinterpreted “SPSO” as	Success: 100% Time: 1 min Error: Hesitated before clicking “SPSO.”	Success: 100% Time: 1 min Error: Brief confusion over “SPSO”	Success: 100% Time: 1 min Error: Misunderstood “SPSO” as	Success: 100% Time: 1 min Error: None.

	unrelated to admin login.		meaning.	system settings.	
2. Login Page	Success: 100% Time: 1.5 min Error: No feedback for an intentionally incorrect login input.	Success: 90% Time: 2 min Error: Entered incorrect credentials and found no error guidance.	Success: 100% Time: 1.5 min Error: None	Success: 90% Time: 2 min Error: Missed error feedback on incorrect credentials.	Success: 100% Time: 1.5 min Error: None.
3. User Homepage After Login	Success: 100% Time: 1.5 min Error: None.	Success: 100% Time: 1.5 min Error: None	Success: 100% Time: 1 min Error: Hesitated finding “New printing job.”	Success: 100% Time: 1.5 min Error: None.	Success: 100% Time: 1.5 min Error: None.
4. Upload Documents Page	Success: 90% Time: 2 min Error: Missed the drag-and-drop zone initially.	Success: 80% Time: 2.5 min Error: Drag-and-drop area unclear; uploaded through “Browse” instead.	Success: 100% Time: 1.5 min Error: None.	Success: 90% Time: 2 min Error: Hesitated to use drag-and-drop.	Success: 100% Time: 1.5 min Error: Progress indicator not immediately noticeable.
5. Select Printer Page	Success: 100% Time: 2 min Error: Briefly confused by similar printer names.	Success: 100% Time: 2 min Error: Printer labels not clear enough for easy selection.	Success: 100% Time: 1.5 min Error: None.	Success: 100% Time: 2.5 min Error: None	Success: 100% Time: 2 min Error: None.

6. Printing Properties Page	Success: 100% Time: 2 min Error: None.	Success: 90% Time: 2.5 min Error: Couldn’t find a reset option to revert changes.	Success: 100% Time: 1.5 min Error: None.	Success: 100% Time: 2 min Error: None.	Success: 90% Time: 2.5 min Error: No reset option caused confusion.
7. Preview Page	Success: 100% Time: 1.5 min Error: Wanted to zoom into the preview.	Success: 80% Time: 2.5 min Error: Couldn’t find an option to edit the document in preview	Success: 100% Time: 1.5 min Error: None.	Success: 100% Time: 1.5 min Error: None.	Success: 90% Time: 2 min Error: Missed zoom and annotation tools.
8. Printing Log Page	Success: 100% Time: 2 min Error: None.	Success: 80% Time: 2.5 min Error: Missed advanced filtering options.	Success: 100% Time: 2 min Error: None	Success: 90% Time: 2.5 min Error: No download option for student logs.	Success: 80% Time: 3 min Error: Couldn’t sort logs by date or location.

5.2.5. Document the feedback from testers.

The survey titled "MVP1 Usability Test" is designed in google form to collect feedback from testers on their experiences using the SSPS software. The primary goal is to understand their satisfaction with various features and tasks, identify areas for improvement, and gather suggestions for enhancements. Testers are asked to rate their satisfaction on a scale from "Very Unsatisfied" to "Very Satisfied" and provide detailed feedback where necessary.

MVP1 usability test

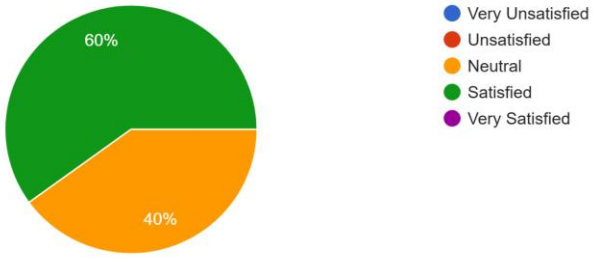
Purpose of the survey:
Thank you for participating in the usability testing of the SPSS software. This survey is designed to capture your thoughts and opinions on various aspects of the software based on the tasks you completed during the test. Your responses will remain confidential and will only be used for research and improvement purposes.

How to complete the survey:

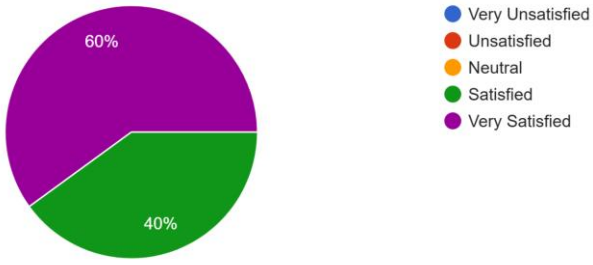
1. Please answer all the questions honestly, based on your experience with the software.
2. Use the rating scale provided (from "Very Unsatisfied" to "Very Satisfied") to rate your satisfaction with specific features and tasks.
3. For questions asking for suggestions or comments, feel free to provide detailed feedback to help us understand your perspective better.

Here are the survey results from all testers:

General Questions 1. How satisfied are you with the ease of navigation in the interface?
5 responses

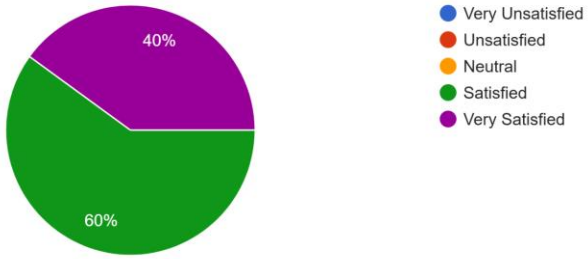


2. How satisfied are you with the overall user experience of the SSPS software?
5 responses



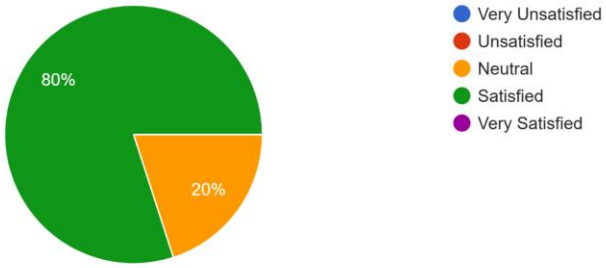
3. How satisfied are you with the intuitiveness of the interface?

5 responses



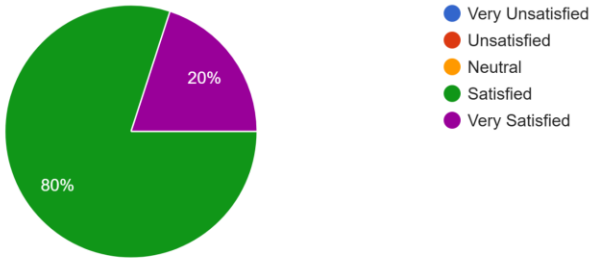
4. How satisfied are you with the clarity of the instructions provided for completing tasks?

5 responses



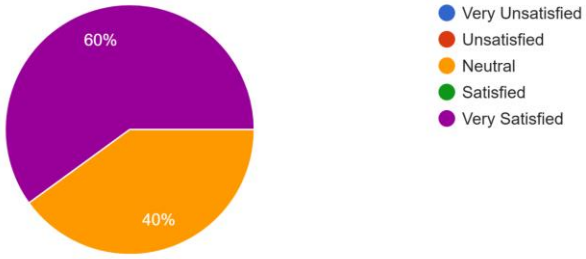
Task-Specific Questions 1. How satisfied are you with the ease of navigating to the student and admin login pages?

5 responses



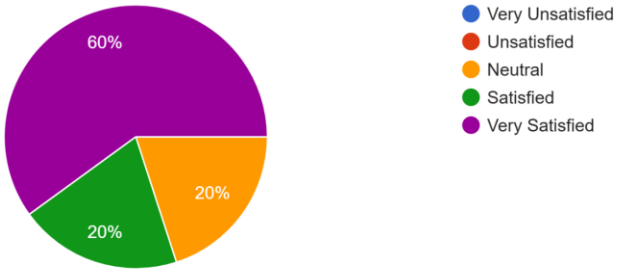
2. How satisfied are you with the ease of logging into the system as a student/SPSO?

5 responses



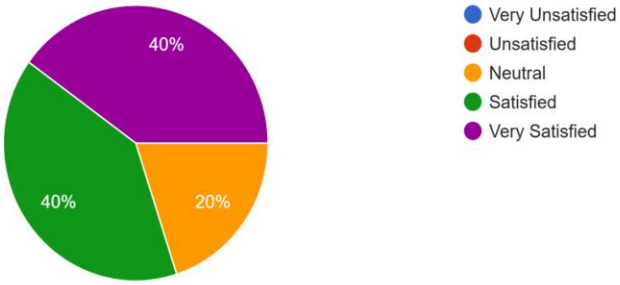
3. How satisfied are you with the ease of uploading, deleting, and re-uploading documents?

5 responses



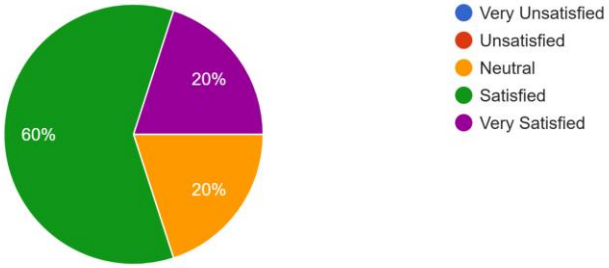
4. How satisfied are you with the system's support for required file formats?

5 responses



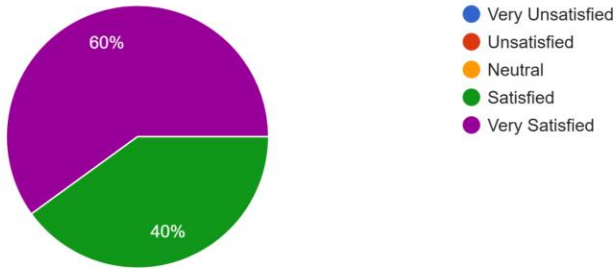
5. How satisfied are you with the process of choosing and re-choosing a printer?

5 responses



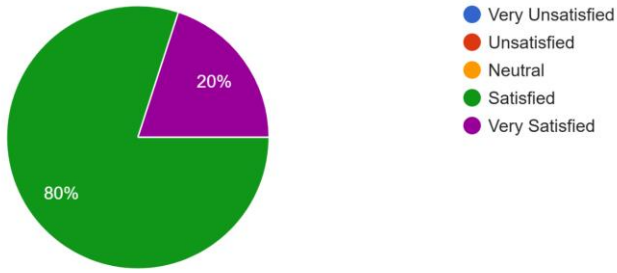
6. How satisfied are you with the process of specifying and re-specifying printing properties?

5 responses



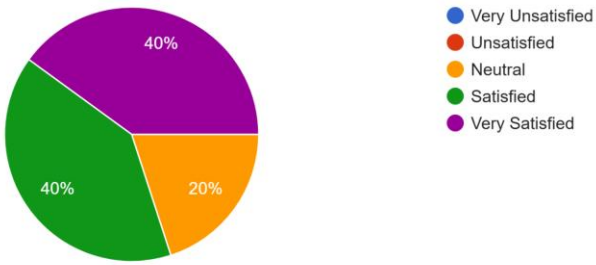
7. How satisfied are you with the usefulness of the document preview feature before printing?

5 responses



8. How satisfied are you with the ease of viewing and navigating through printing logs?

5 responses



SSPS Feedback: What the most useful and What needs improvement

What aspects of the SSPS software did you find most useful?

3 responses

- File upload process was quick and smooth
- Logs were clear and easy to navigate for both students and admins
- Dropdown menus were responsive and easy to use

What areas of the software need improvement?

3 responses

- Error messages were not descriptive enough, leaving users unsure of how to fix issues like login failures
- Page transitions, while functional, felt slightly slower on lower-spec devices, which could lead to frustration
- Add a “Remember Me” checkbox for returning users in login page.