Software Requirements Specification

for

<A smart printing service for students at HCMUT>

Version 3.1 approved

Prepared by:

1. Trần Tuấn Kiệt — 2252410
2. Trương Tuấn Kiệt — 2252411
3. Văn Duy Anh — 2252045
4. Vương Khang — 2250008
5. Trương Quốc Thuận — 2252790
6. Võ Phú Thịnh – 2252782

Department of Software Engineering

Faculty of Computer Science and Engineering

Ho Chi Minh City University of Technology – VNU-HCM

<09/11/2024>

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
First version	22/09/2024	Initial Release	Version 1.0
Task 1 (Final)	01/10/2024	Add part 2. Finalize for Task 1 requirements.	Version 1.1
Task 2 (Final)	27/10/2024	Finalize for Task 2 requirements.	Version 2.1
Task 2 (Final)	09/11/2024	Finalize for Task 3 requirements.	Version 3.1

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

1.1 Domain Context

The HCMUT Smart Printing Service (HCMUT-SSPS) aims to facilitate students by providing a centralized and convenient solution for printing documents at various locations across the university campus. The system allows students to upload files, select printing preferences, and manage their printing needs efficiently. This service ensures that students have access to reliable and modern printing solutions, thus enhancing their academic productivity.

1.2 Stakeholders and Needs

The primary stakeholders include:

1.2.1 Students:

Students need a reliable and convenient system to print documents for their studies. First, the system must be accessible through both web and mobile apps, allowing students to manage their printing tasks from various devices. In addition, students also require an user-friendly interface for uploading, choosing printers and printing configuration. Moreover, students want to track their usages and an online payment method to purchase additional printing credits. Finally, the students' personal information and printing activity must be protected by the system via HCMUT_SSO authentication.

1.2.2 SPSO (Student Printing Service Officer):

The needs of the Student Printing Service Officer (SPSO) are centered around effective management and oversight of the printing services offered to students. The SPSO requires a comprehensive dashboard that allows for real-time monitoring of printer statuses, usage statistics, and student printing activity. They need the ability to configure system settings, such as adjusting the default number of pages allocated to students and managing permitted file types for printing. Additionally, the SPSO seeks tools to generate detailed reports on printing activities, both monthly and annually, to analyze usage trends and make data-driven decisions. Ensuring the smooth operation of printers, including adding, enabling, or disabling them as necessary, is also a critical need, as is maintaining a balance between resource availability and student demand.

1.2.3 System Administrator:

Requires full control over system configuration, including managing user accounts, overseeing printer status, and ensuring system security. The System Administrator needs the ability to view system logs, update system settings, and manage backup and recovery processes to minimize downtime and maintain operational efficiency.

1.3 Benefits of the System

The HCMUT-SSPS will offer several benefits:

1.3.1 For Students:

The Student Smart Printing Service (HCMUT_SSPS) provides a lot of convenience, printing activity control, and security. Students can print from any location with multiple printers available across various buildings and manage their printing needs on the go using either the web-based or mobile app, offering flexible and convenient access to services, even remotely. Moreover, students have access to their own detailed printing history, allowing them to track what, when, and how much they printed. This helps in managing their printing budget and understanding their usage patterns. Finally, using the HCMUT_SSO authentication system ensures that only authorized users (the students themselves) can access their printing accounts, protecting students' personal information and preventing unauthorized use of printing quotas.

1.3.2 For SPSO:

The system allows the SPSO to easily add, enable, or disable printers remotely, ensuring efficient control over the university's printing infrastructure.. Moreover, the SPSO can also view the printing history (log) of all students or a student for a time period (date to date) and for all or some printers and managing other configurations of the system such as changing the default number of pages, the dates that the system will give the default number of pages to all students, the permitted file types accepted by the system. Therefore, SPSO can know the amount of pages that a student can print in that time and provide pages if they use the feature Buy Printing Pages of the system. At the end of each month and each year, the system will automatically generate and store the reports of the use of the printing system that can be viewed by SPSO, so SPSO can easily manage and recognize mistakes of the HCMUT_SSPS.

1.3.3 For System Administrator:

The HCMUT-SSPS provides the System Administrator with a centralized platform to manage all aspects of the printing infrastructure efficiently. Administrators can oversee and control user accounts, manage printer configurations such as locations and availability, and ensure the security of the system. The ability to monitor real-time printer statuses allows for quick detection and resolution of issues, minimizing downtime and ensuring continuous operation. Regular backups and comprehensive system logs enhance system reliability,

enabling swift troubleshooting when needed. Additionally, the system generates detailed financial and usage reports, offering valuable insights into resource allocation and system performance, which supports long-term operational planning.

1.4 Functional Requirements

1.4.1 Students:

- Students shall be able to upload documents for printing, select a specific printer and customize properties.
- Students shall be able to purchase additional printing credits via online payment.
- Students shall be able to view their printing log (including student ID, printer ID, file name, printing start and end time, number of pages for each page size) for a time period together with a summary of the number of printed pages for each page size, and the available printing pages.
- The system must not allow students to print some number of pages when it exceeds their account balance.
- The system must validate the uploaded document's format and only allow printing for supported file types, as defined by the Student Printing Service Officer (SPSO).
- If a student uploads file types that are not permitted, the system must display an error message to the student notifying them that the file types are not allowed.
- The system must integrate with the HCMUT_SSO authentication service, ensuring that only authenticated students can access the printing services.

1.4.2 The SPSO:

- SPSO can manage printers such as add/enable/disable a printer.
- SPSO can view the reports of the using of the printing system which are generated automatically at the end of each month and each year and are stored in the system
- SPSO centralized log system for reviewing student print activities.
- SPSO must be able to adjust system configurations such as default number of pages, permitted file type, and default page allocation dates.
- SPSO must be authenticated via the HCMUT_SSO service to access system functionalities.

1.4.3 System Administrator:

- The system must allow the System Administrator to create, update, or delete user accounts.
- The system must allow the System Administrator to update printer details such as the printer name, location, and availability.
- The system must notify the System Administrator when printers run out of paper or ink.

- The system must allow the System Administrator to generate and export detailed financial reports on system usage.
- The system must allow the System Administrator to monitor the real-time status of all connected printers, including errors and paper jams.
- The system must enable the System Administrator to configure automated printer maintenance schedules and receive reminders for tasks like ink replacement.

1.5 Non- Functional Requirements

- The system must be secure, requiring authentication via the HCMUT SSO system.
- The system must be available 99% of the time during campus hours.
- The web and mobile apps must provide a responsive and user-friendly experience.
- The system must ensure that documents are uploaded and processed for printing within 5 seconds to ensure smooth operation.
- The system must be accessible through both web and mobile apps.
- The system must handle a minimum of 100 simultaneous users without significant performance degradation.

2. Use-case Diagrams (1.3)

2.1 Use-case Diagram for the Whole System

The following use case diagrams and corresponding descriptions are developed under the assumption that all users are required to authenticate via the HCMUT-SSO system before proceeding to the system's homepage. Upon successful login, the system will automatically retrieve and verify the user's identity, including personal information and designated role—whether as a student or a Smart Printing Service Office (SPSO) staff member.

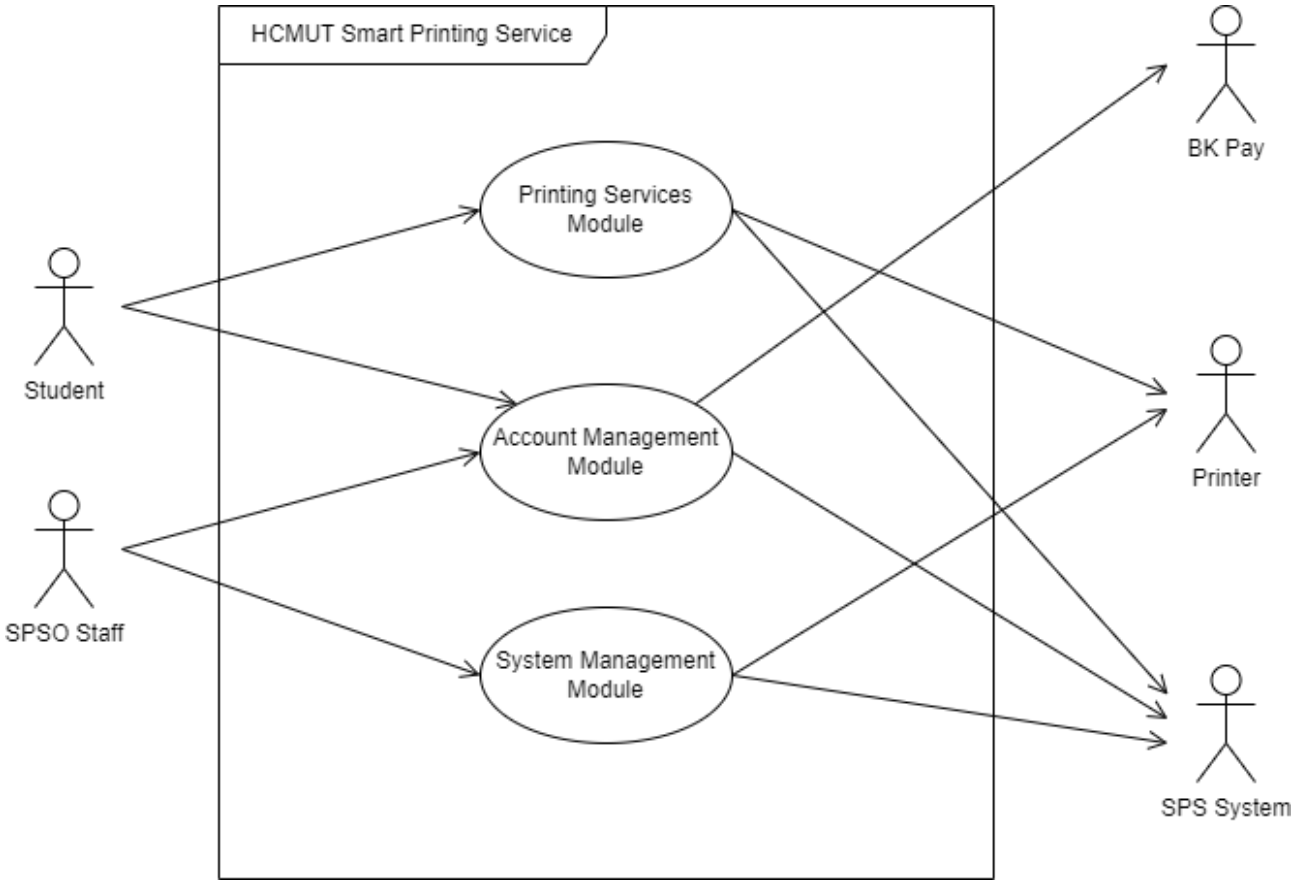


Figure 1: Use-case diagram for the HCMUT Smart Printing Service System.

2.2 Use-case Diagram for <Printing Services> Module

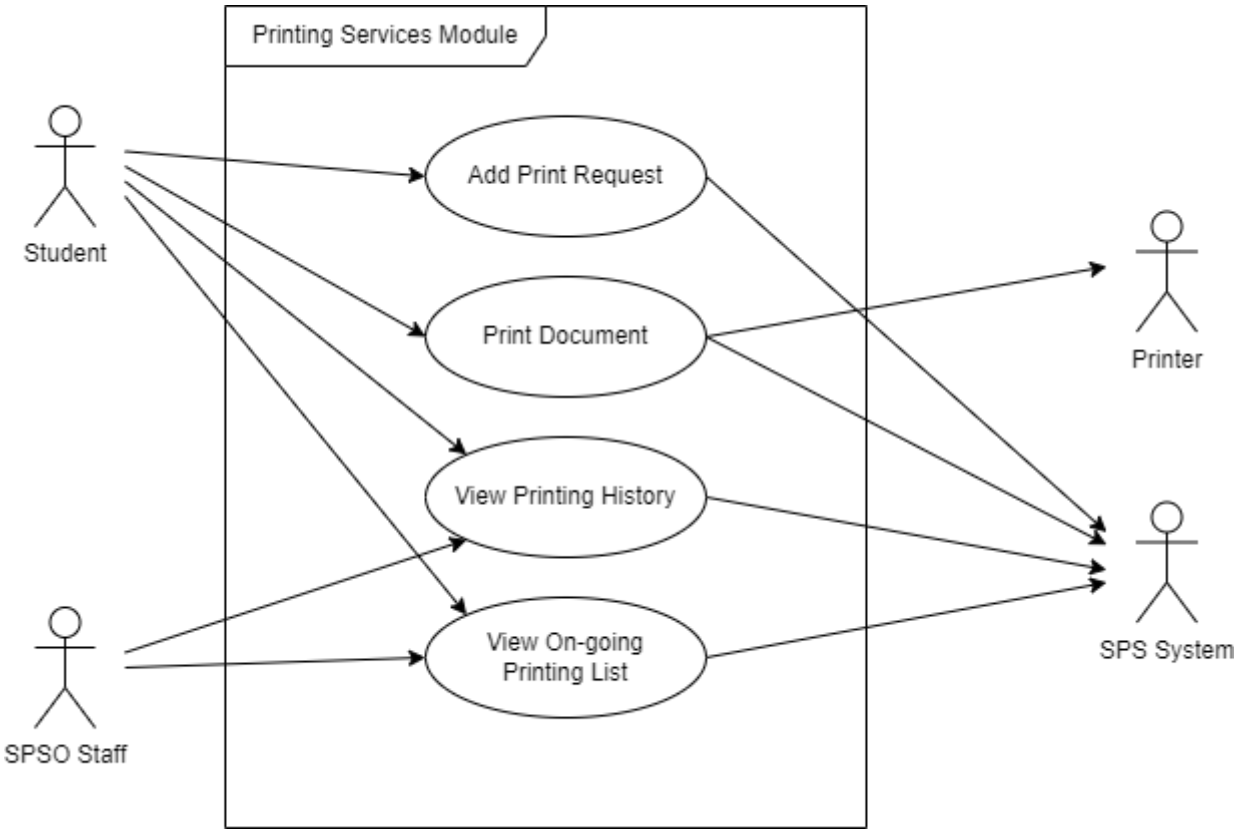


Figure 2: Use-case diagram for the <Printing Services> Module.

2.3 The Details of Use cases in the <Printing Services> Module.

1. Use-case <Add Printing Request>

ID and Name:	**UC-1. Add Printing Request**		
Created by:	Truong Tuan Kiet	**Date Created:**	28/09/2024
Primary Actor:	Student	**Secondary Actors:**	SPS System
Description:	The student uploads a document for printing, selects a printer, and customizes the printing properties (e.g., paper size, number of copies, one-/double-sided printing).		
Trigger:	The student wants to print a document and chooses “Request printing”.		
Preconditions:	**PRE-1.1.** The student is authenticated via HCMUT_SSO. **PRE-1.2.** The SPS System’s database is online and available.		
Post-conditions:	**POST-1.1.** A print request is added to the waiting list with all the required properties (document, printer, printer setting,…) **POST-1.2.** The student is informed that the print request is successfully added to the queue.		
Normal Flow:	**1.0. Submit a Printing Request.** 1. The student navigates to the "Add Print Request" section. 2. The system displays the "Upload Document" interface. 3. The student selects and uploads a document file. 4. The system verifies the file type against the permitted file types: • If valid, the system accepts the file. • If invalid, the system displays an error message and prompts the student to upload a valid file type (see 1.1). 5. The system displays a list of available printers for selection. 6. The student selects a printer from the list. 7. The student configures printing properties (e.g., paper size, one-sided/double-sided, number of copies). 8. The student confirms the print request. 9. The system adds the print request to the waiting list and displays a confirmation message. 10. The student can view the status of their print request in the queue.		
Alternative	**1.1. Invalid File Type.**		

Flows:	4.a. If the uploaded file is not of a permitted type, the system displays an error message. 4.b. The student is prompted to upload a valid file and repeat the process.
Exceptions:	None

2. Use-case <Print Document>

ID and Name:	**UC-2. Print Document**		
Created by:	Văn Duy Anh Trương Quốc Thuận	**Date Created:**	28/09/2024
Primary Actor:	Student	**Secondary Actors:**	Printer SPS System
Description:	The system processes a student's print request by checking their page balance and sending the document to the selected printer if sufficient. It logs the print details and notifies the student upon completion. If the balance is insufficient or issues like printer malfunctions occur, the student is alerted to resolve the issue or try again.		
Trigger:	• The student decides to print a document from wait list.		
Preconditions:	**PRE-2.1.** The student is authenticated via HCMUT_SSO. **PRE-2.2.** The SPS System's database is online and available. **PRE-2.3.** The selected printer is available and functioning properly.		
Post-conditions:	**POST-2.1.** The student's document is submitted to the printer. **POST-2.2.** The student receives a notification that their document has been successfully printed.		
Normal Flow:	**2.0. Print Document.** 1. The system retrieves the next print request from the waiting list. 2. The system checks the student's account balance to ensure they have sufficient pages for the print job: • If sufficient, the system proceeds to the next step. • If insufficient, the system displays an error message and cancels the print request (see 2.1). 3. The system sends the document to the selected printer. 4. The printer begins printing the document. 5. The system logs the printing action, including details such as student ID, printer ID, number of pages printed, paper size… 6. The student receives a notification that their document has been successfully printed.		
Alternative Flows:	**2.1. Insufficient Page Balance** 2.a. If the student's account balance is insufficient, the system displays an error message. 2.b. The student is notified that they need to buy additional credits		

	before proceeding with the print job. 2.c. The print request is removed from the queue or marked as pending until the student has enough credits.
Exceptions:	**2.0.E1. Printer Malfunction or Unavailability:** If the selected printer encounters a malfunction or becomes unavailable before or during printing, the system will display an error, and the print request may either be marked as pending or the student is notified to reselect another printer. **2.0.E2. Network/System Failure:** If there is a network or system failure, the printing process is halted, and the student is notified to retry later or select an alternative action.

3. Use-case <View Printing History>

ID and Name:	UC-3. View Printing History		
Created by:	Vương Khang Võ Phú Thịnh	**Date Created:**	28/09/2024
Primary Actor:	Student SPSO Staff	**Secondary Actor:**	SPS System
Description:	The SPSO can view the printing history (log) of all students or a student for a time period (date to date) and for all or some printers. The SPSO can specify the desired Student by entering Student ID, the desired time period and the desired printers by entering Printer ID(s). Moreover, The Student can also view his/her printing log for a specified time period with a summary of the number of printed pages for each page size. The Student specifies the desired time period.		
Trigger:	The user selects the option to view printing history from the system's interface.		
Preconditions:	**PRE-3.1.** The user is authenticated via HCMUT_SSO. **PRE-3.2.** The SPS System's database is online and available.		
Post-conditions:	***Student:*** • **POST-3.1.1.** The student's printing history (only their own data) is displayed for the selected time period. • **POST-3.1.2.** The system has restricted access to ensure the student can only view their own records. ***SPSO staff:*** • **POST-3.2.1.** The SPSO can view the printing history for all students, with options to filter by date,student and printers. • **POST-3.2.2.** The system provides full access to the printing history for operational management.		
Normal Flow:	**3.0. View Printing History.** 1. The system checks whether the user's role is student or SPSO staff 2. If the actor is a Student: the system asks the user to specifies the desired time period 3. System displays the printing log of the Student with a summary of the number of printed pages for each page size.		

Alternative Flows:	**3.1. The user is a SPSO Staff Member.** 2a. If the user is a SPSO staff member, the system asks the user to specify the information of the query 2b. The staff member chooses to query on printers or on students. 2b1. If the staff member chooses "printers", the system asks the users to choose specific printers from the list of printers and time period for the query. 2b2. If the staff member chooses "students", the system asks the users to input students' IDs and time period for the query. 2c. System retrieves the printing log based on the query's specification. 2d. The printing log is displayed on the screen.
Exceptions:	**3.0.E1. The system retrieves an empty history:** A notification displayed on screen to notify that there is no printing history to be displayed. **3.0.E2. Invalid input selection:** The actor enters invalid input (invalid date range, student ID, or printer ID(s)). The system displays a message notifying the error for the SPSO to adjust. The use case returns to step 3.

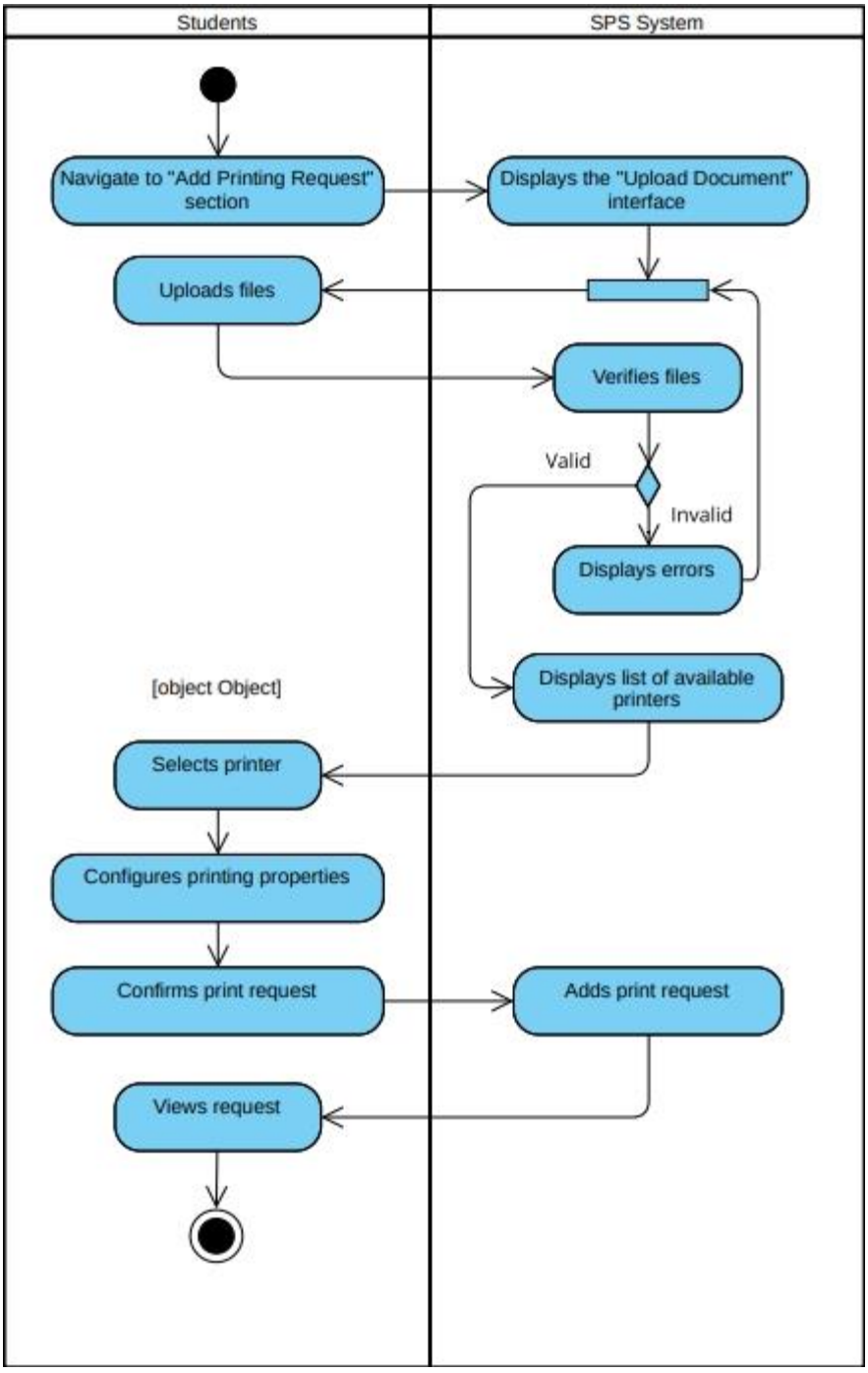
4. Use-case <View On-going Printing Requests>

ID and Name:	**UC-4. View On-going Printing Requests**		
Created by:	Tran Tuan Kiet	**Date Created:**	28/09/2024
Primary Actor:	Student, SPSO Staff	**Secondary Actors:**	SPS System
Description:	The system displays a list of on-going printing requests after authenticating the user. If the user is a student, their personal list is shown. If the user is SPSO staff, they can query the list by printer or student. The system then retrieves and displays the relevant list. If no requests are found, a notification is shown.		
Trigger:	The user selects the option to view on-going printing requests from the system's interface.		
Preconditions:	**PRE-4.1.** The user is authenticated via HCMUT_SSO. **PRE-4.2.** The SPS System's database is online and available.		
Post-conditions:	**POST-4.1.** The list of on-going printing requests is displayed.		
Normal Flow:	**4.0. View On-going Printing Requests.** 4. System checks whether the user's role is student or SPSO staff (see 4.1). 5. If the user is a student, the system retrieves the student's on-going printing list from the database. 6. A list of on-going printing requests is displayed on the screen.		
Alternative Flows:	**4.1. The user is a SPSO Staff Member.** 2a. If the user is a SPSO staff member, the system asks the user to specify the information of the query. 2.b. The staff member chooses to query on printers or on students. 2.b.1. If the staff member chooses "printers", the system asks the users to choose specific printers from the list of printers and time period for the query. 2.b.2. If the staff member chooses "students", the system asks the users to input students' IDs and time period for the query. 2.c. System retrieves the list based on the query's specification. 2.d. The list of on-going printing requests is displayed on the screen.		
Exceptions:	**4.0.E1. The system retrieves an empty list:** A notification displayed on screen to notify that there are no on-going printing requests.		

3. Task 2: System modelling.

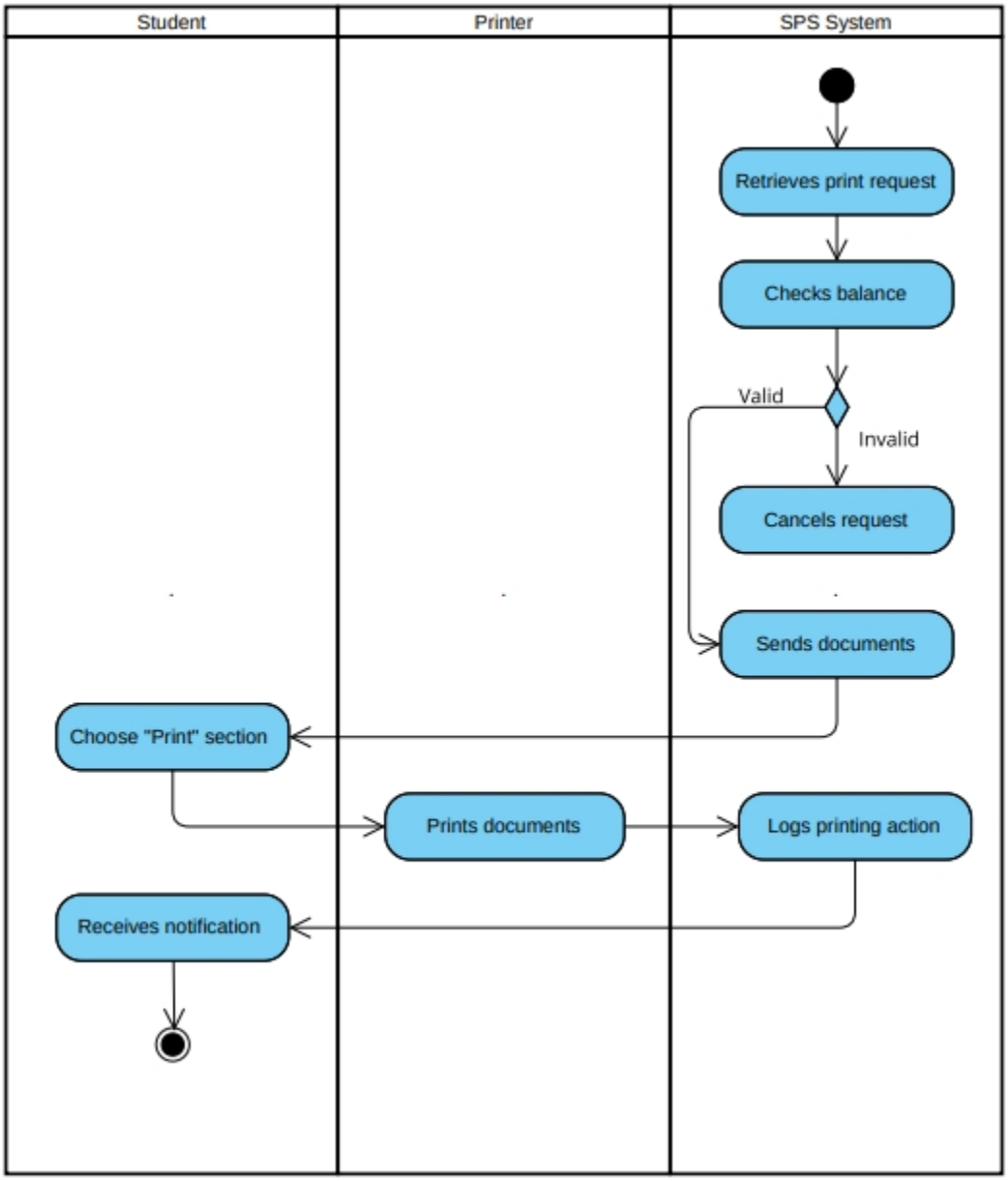
3.1 Task 2.1: Activity diagrams.

Use Case 1: Add Printing Request



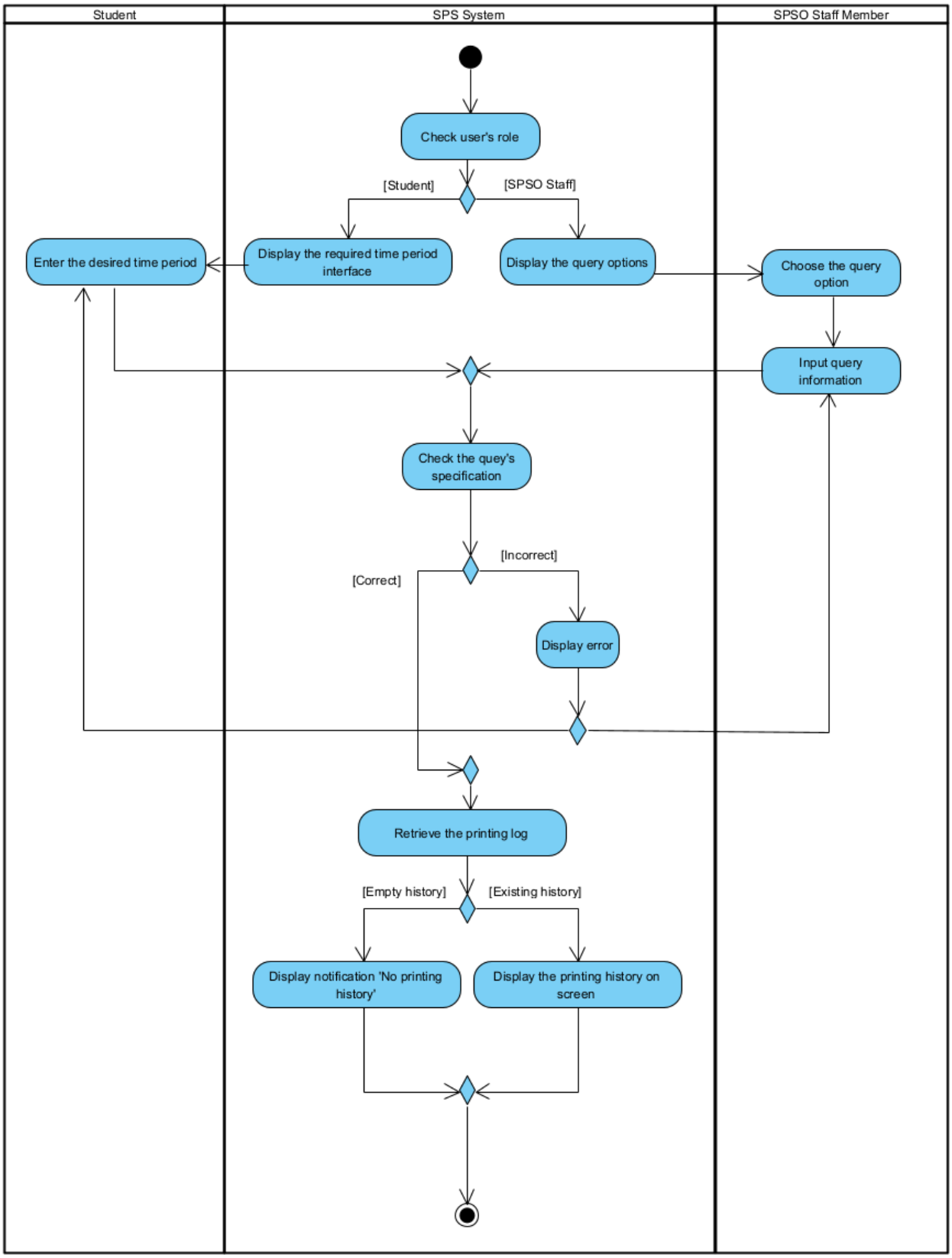
The activity diagram for the use case "Add Printing Request" illustrates the sequence of actions taken by the student while submitting a print request through the SPS System. The process begins with the student navigating to the "Add Print Request" section, where the interface for uploading documents is displayed. Upon uploading a document, the system verifies the file type. If the file type is valid, it proceeds to show a list of available printers. The student then selects a printer and customizes the printing properties, such as paper size and number of copies. After confirming the print request, the system adds it to the waiting list and displays a confirmation message. Finally, the student can check the status of their print request in the queue. If an invalid file type is uploaded, the system prompts the student to upload a valid file, ensuring the process is user-friendly and efficient.

Use Case 2: Print Document



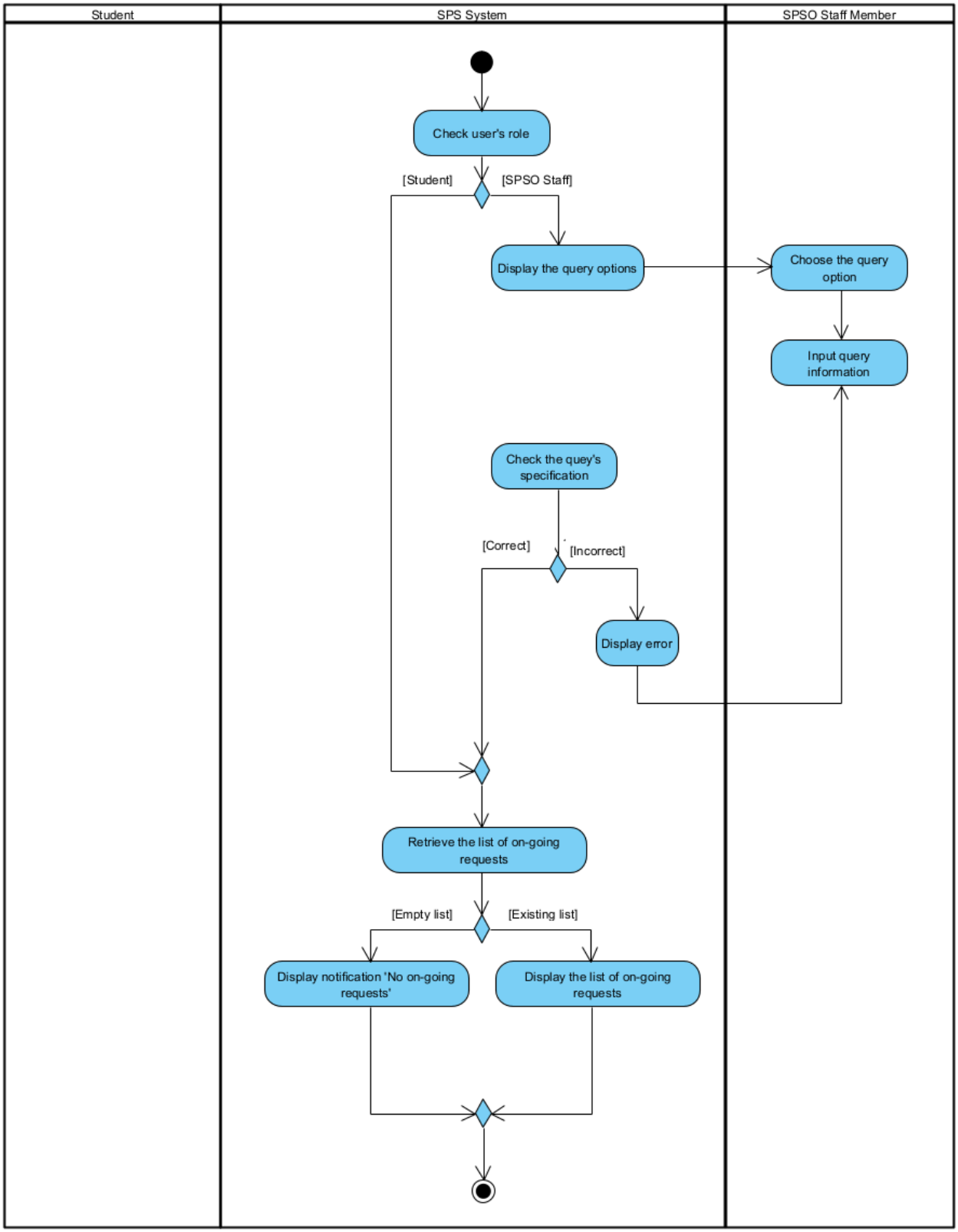
The activity diagram for the use case "Print Document" outlines the steps involved in processing a student's print request through the SPS System. Initially, the system retrieves the next request from the waiting list and checks the student's account balance to ensure there are sufficient pages available for printing. If the balance is sufficient, the document is sent to the selected printer, which then begins the printing process. The system logs relevant details, such as student ID and number of pages printed, and subsequently notifies the student that their document has been successfully printed. If the account balance is insufficient, an error message is displayed, prompting the student to purchase additional credits. The request may then be removed from the queue or marked as pending until the student resolves the balance issue.

Use Case 3: View Printing History



The activity diagram for the use case “View Printing History” shows the interactions between Student, SPSO Staff Member and SPS System in the “View Printing History” process. At initial, the process starts when the user selects the option to view the printing history. Next, The SPS System will check the user’s role to give the appropriate interface. If the user is Student, the user will enter his/her desired time period for viewing printing history. If the user is SPSO Staff, the user will choose the query option (query on Printers or Student); Based on their choice, the user will next input the query’s specification (Printers’ IDs with Printers or Student IDs with Students and the time period). At the next step, the system will check the query’s specification. If there are errors, the system will display Errors notification and return to the interface for users to enter input again. After that, the system will retrieve the printing history from database. If the system retrieves an empty log, the system will display notification on screen to notify that there is no printing history to be displayed. If the printing log exists, it will be displayed on screen.

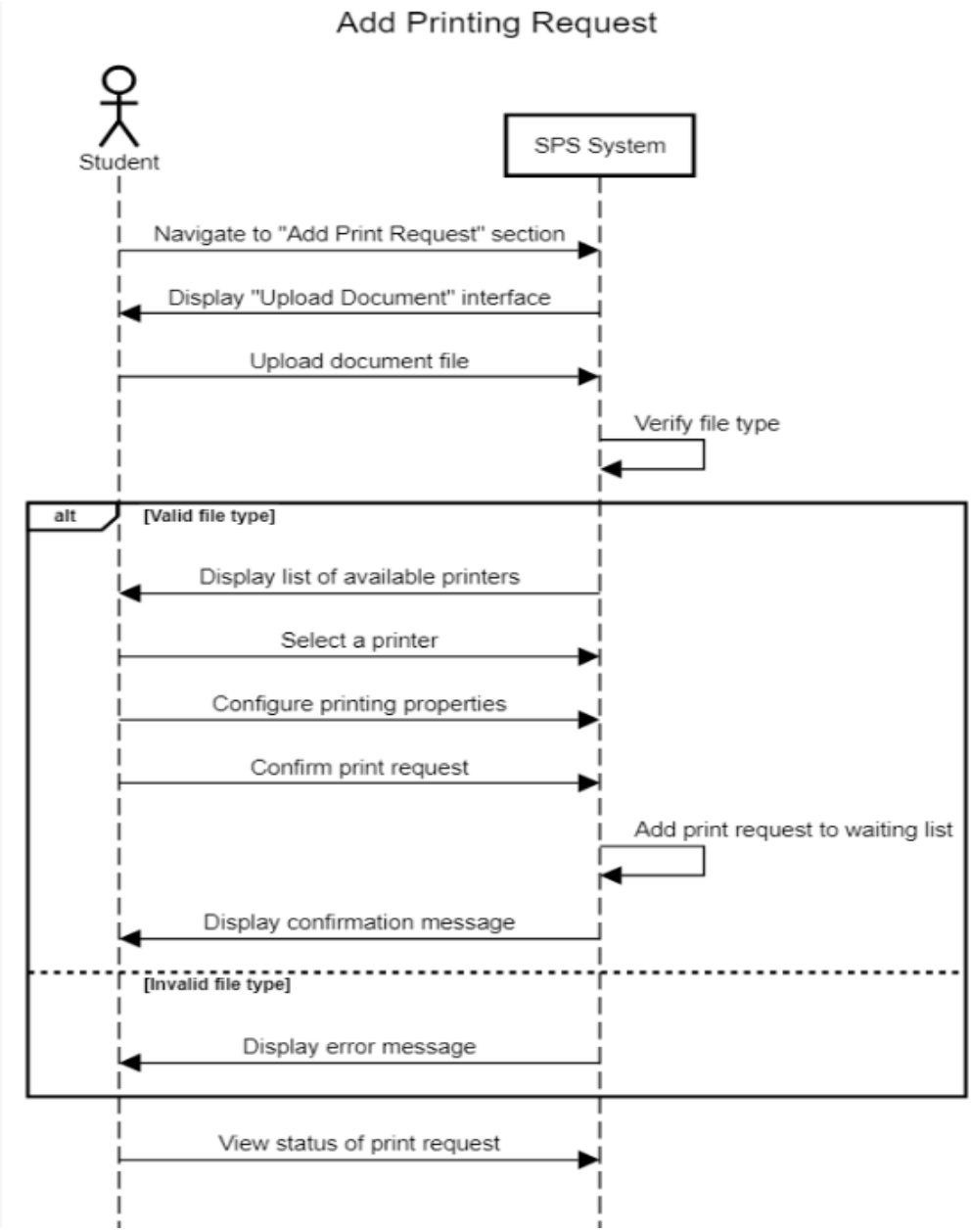
Use Case 4: View On-going Printing Requests



The activity diagram for the use case "View On-going Printing Requests" outlines the process by which a student or SPSO staff member retrieves the list of ongoing printing requests through the SPS System. At initial, the process starts when the user selects the "View On-going Printing Requests" section. Next, The SPS System will check the user's role to give the appropriate interface. If the user is Student, the system will automatically retrieve the list based on his/her Student ID as default. If the user is SPSO Staff, the user will choose the query option (query on Printers or Student); Based on their choice, the user will next input the query's specification (Printers' IDs with Printers or Student IDs with Students and the time period). At the next step, the system will check the query's specification. If there are errors, the system will display Errors notification and return to the interface for users to enter input again. After getting query's information, the system will retrieve the on-going printing list from database. If the system retrieves an empty list, the system will display notification on screen to notify that there is no printing list to be displayed. If the list of on-going printing requests exists, it will be displayed on screen.

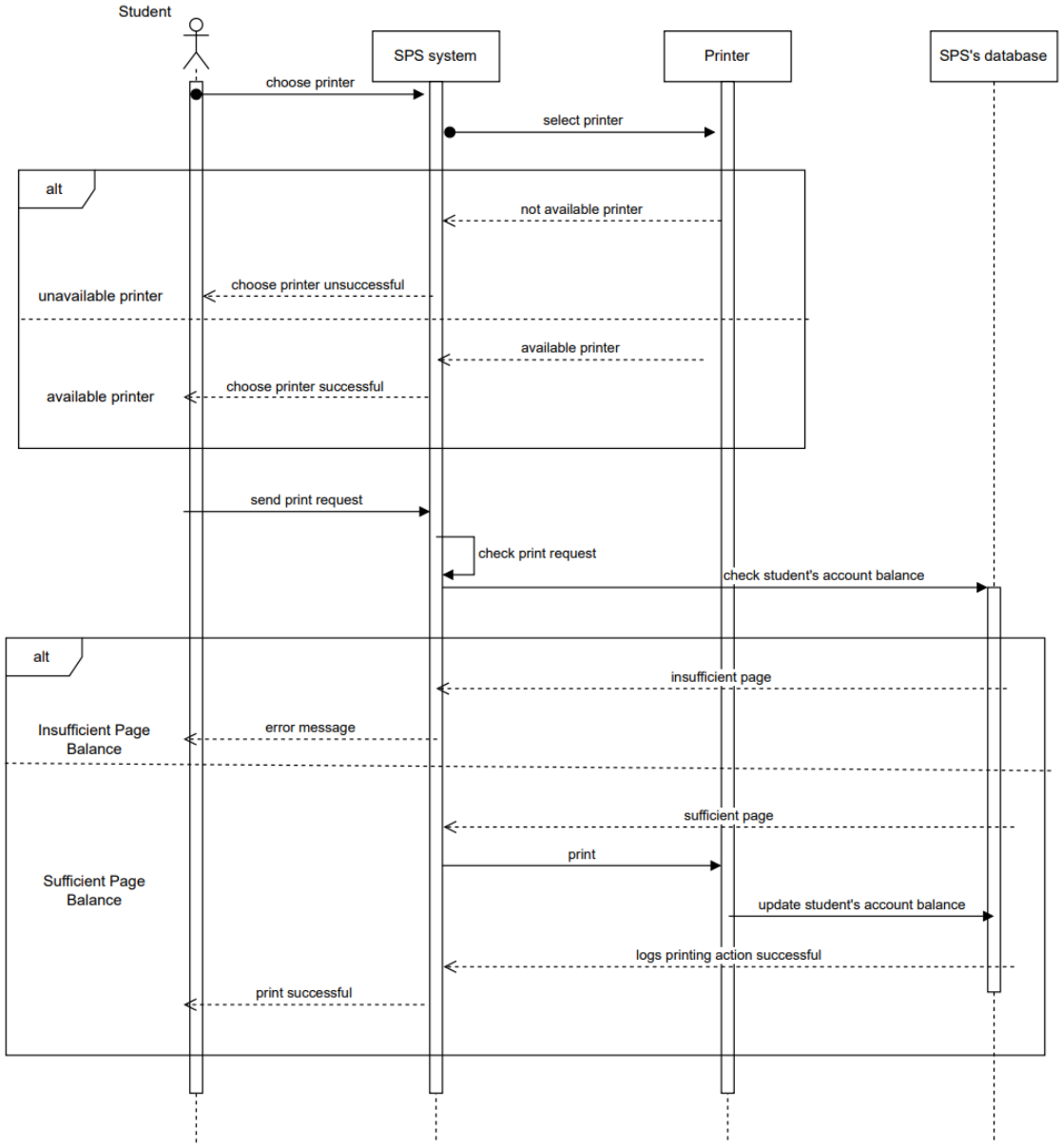
3.2 Task 2.2: Sequence Diagrams.

Add Printing Request



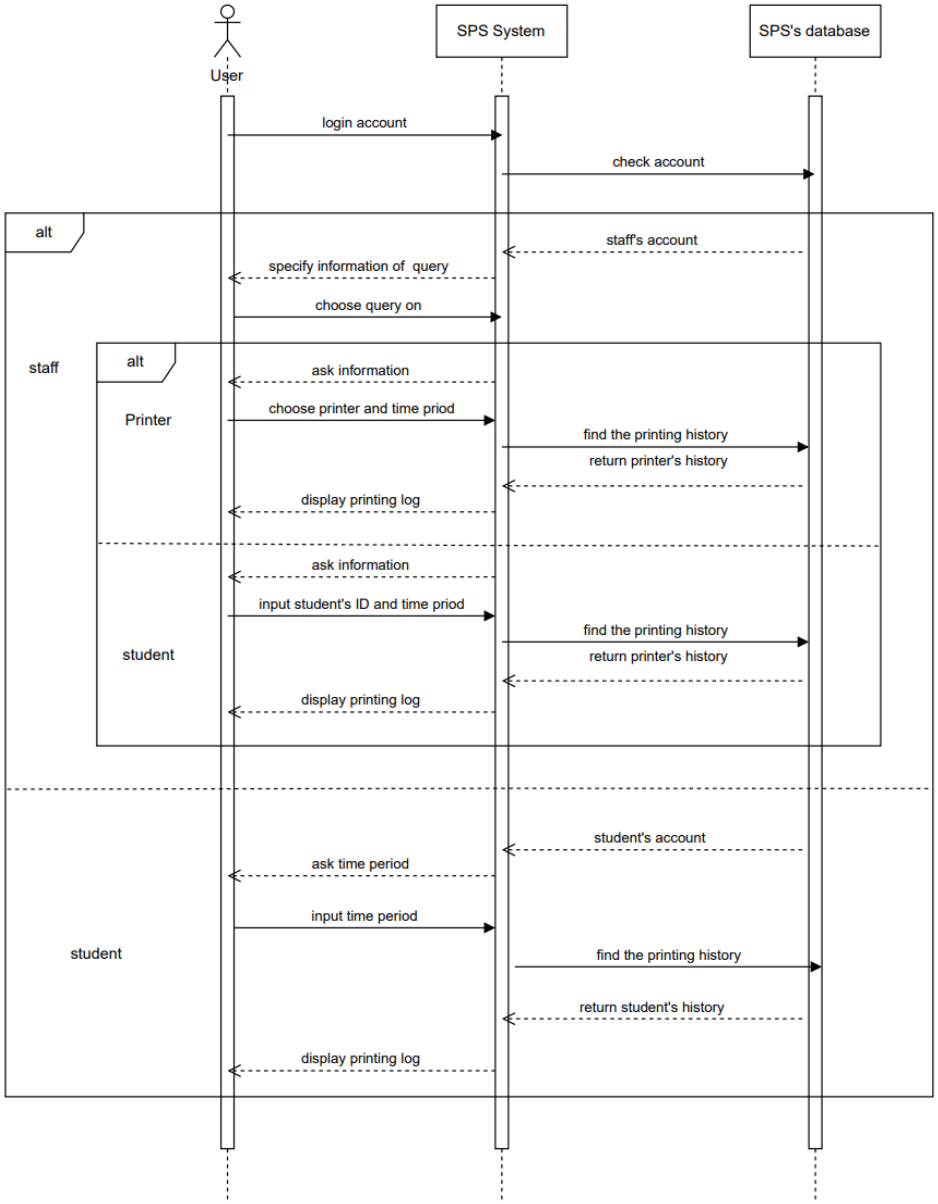
In this sequence diagram, a student initiates a printing request by navigating to the "Add Print Request" section on the SPS System. The system displays the "Upload Document" interface, where the student uploads a document. The system verifies the file type. If valid, the system presents a list of available printers; otherwise, an error message is shown. The student selects a printer, configures printing properties, and confirms the request. The system then adds the request to the waiting list, displays a confirmation message, and allows the student to view the request's status in the queue.

Print document



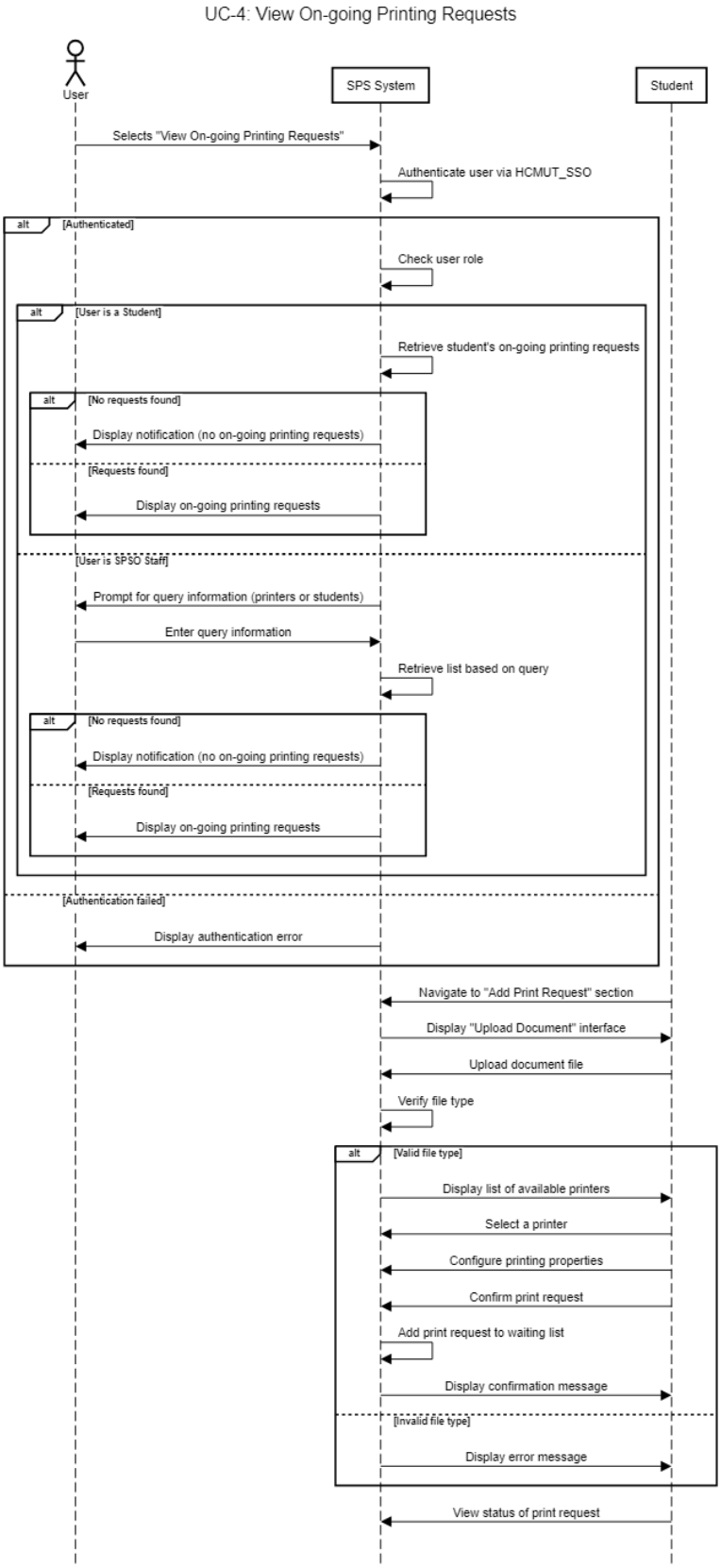
In the sequence diagram, the process begins with the student selecting a printer from the SPS system. The system then attempts to allocate the chosen printer. If the printer is unavailable, the SPS system notifies the student with an "unsuccessful" message, otherwise, it confirms the printer is available with a "successful" message. The student proceeds by sending a print request, which prompts the SPS system to check the request and the student's account balance from the database. If the student has insufficient balance, the system returns an error message indicating the issue. If the balance is sufficient, the system instructs the printer to print the document, updates the student's balance in the database, logs the successful print action, and notifies the student that the print request was completed successfully.

View History



A user will login into the SPS system. The SPS system checks the user's account from the database. If the user is staff, the system asks the user to specify the information of the query . Then the staff member chooses to query on printers or on students. If the staff member chooses "printers", the system asks the users to choose specific printers from the list of printers and time period for the query. If the staff member chooses "students", the system asks the users to input students' IDs and time period for the query. System retrieves the printing log based on the query's specification. The printing log is displayed on the screen. If the actor is a Student: the system asks the user to specifies the desired time period. System displays the printing log of the Student with a summary of the number of printed pages for each page size.

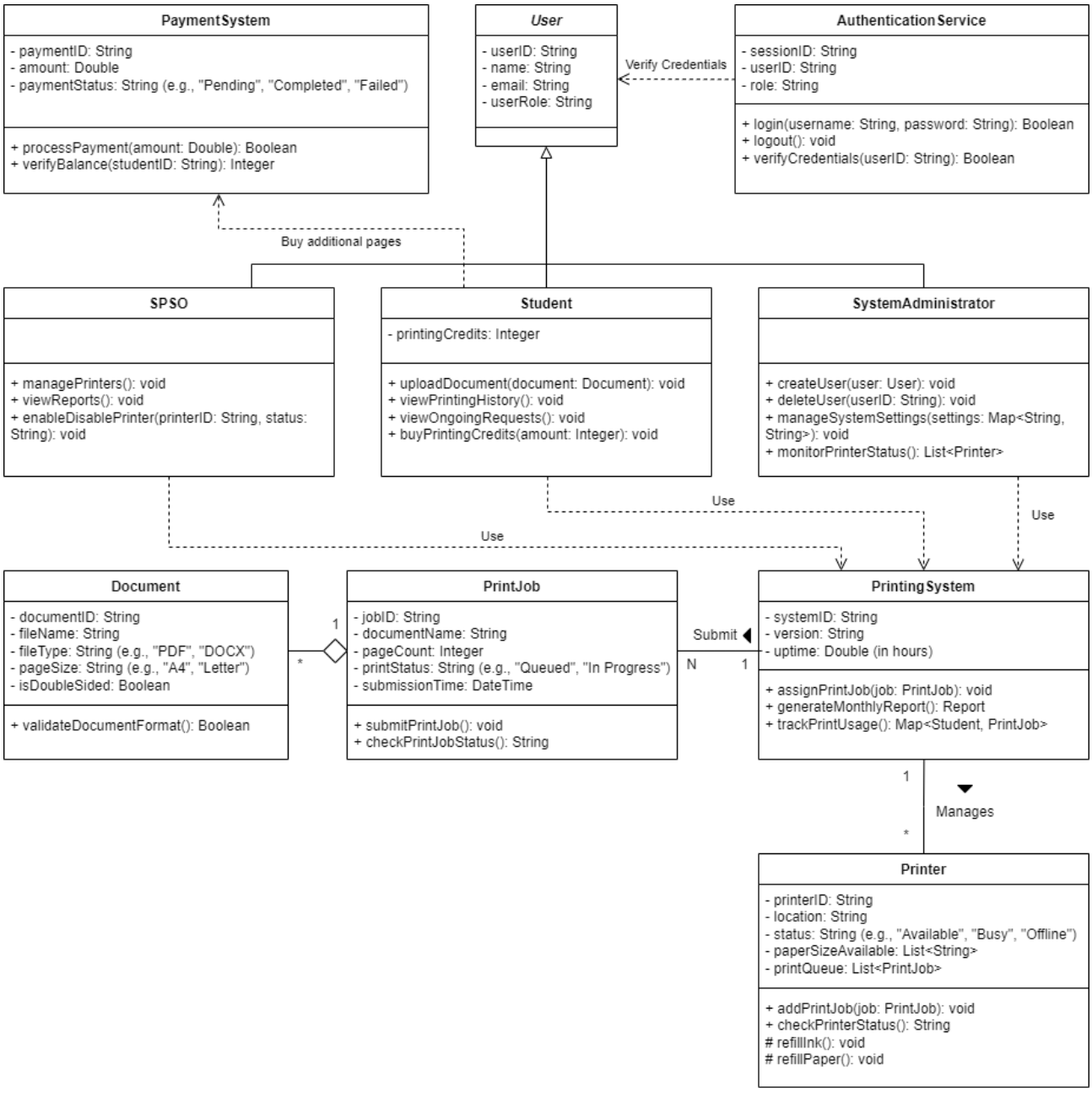
View On-going Printing Requests



This sequence diagram illustrates the process of a user viewing on-going printing requests in the SPS System. The user initiates the process by selecting the "View On-going Printing Requests" option. The system authenticates the user via HCMUT_SSO. If authenticated, the system checks the user's role.

For a student, the system retrieves and displays their personal list of on-going printing requests. If no requests are found, a notification is shown. For SPSO staff, the system prompts for query information (printers or students), retrieves the list based on the query, and displays it. If no requests are found, a notification is displayed. If authentication fails, an error message is shown.

3.3 Task 2.3: Class Diagram.



The class diagram represents the components and interactions of the Student Smart Printing Service (HCMUT_SSPS) for HCMUT students. It shows key entities, including *User*, *Student*, *SystemAdministrator*, *SPSO* (Student Printing Service Officer), *Document*, *PrintJob*, *Printer*, *PaymentSystem*, *AuthenticationService*, and *PrintingSystem*. Each entity has distinct roles and responsibilities in the printing service system.

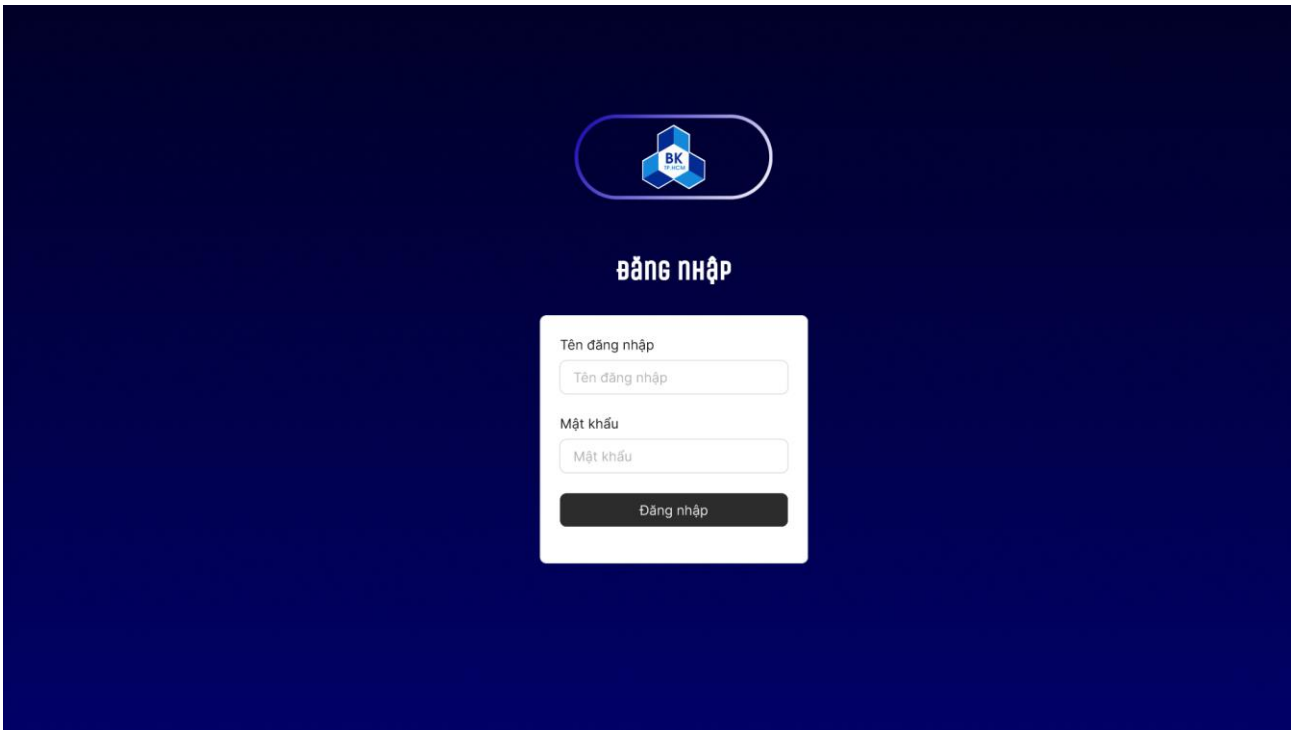
The *User* class is a generalization, with *Student* and *SystemAdministrator* as specific types of users. This setup allows for role-based interactions, where students can manage their printing needs, and administrators manage system settings and monitor logs.

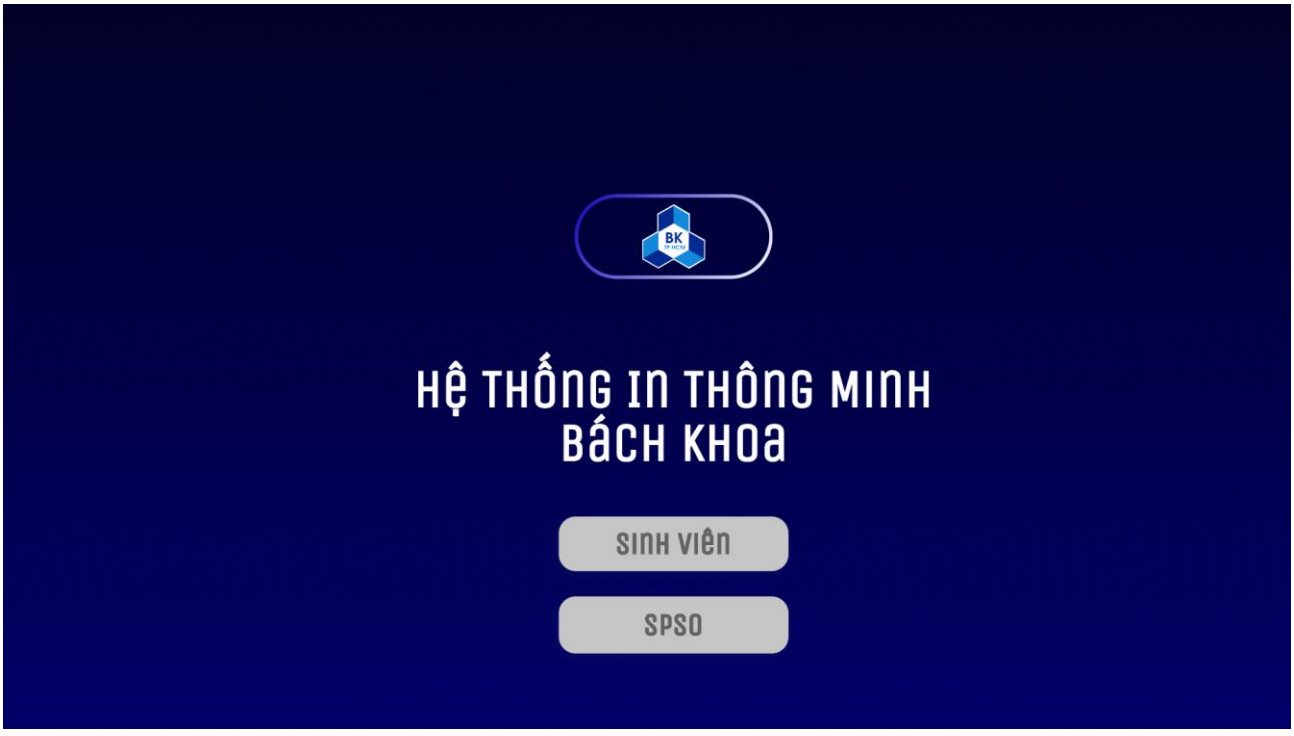
- *AuthenticationService* ensures that only authenticated users can access the system.
- The *SPSO* class is responsible for managing printers, configuring permitted file types, and viewing printing logs, which helps maintain an organized and secure printing service.
- The *Document* class represents the document files uploaded by students, and *PrintJob* manages details like page count, printing properties, and print status, linking to the *Document* and *Printer* classes.
- *PaymentSystem* enables students to purchase additional printing pages if needed.
- The *PrintingSystem* class likely coordinates the entire printing process and links with multiple printers around the campus.

This class diagram provides a clear, modular structure for HCMUT_SSPS, supporting scalability, maintenance, and a secure user experience for students and administrators alike.

3.4 Task 2.4: User Interface.

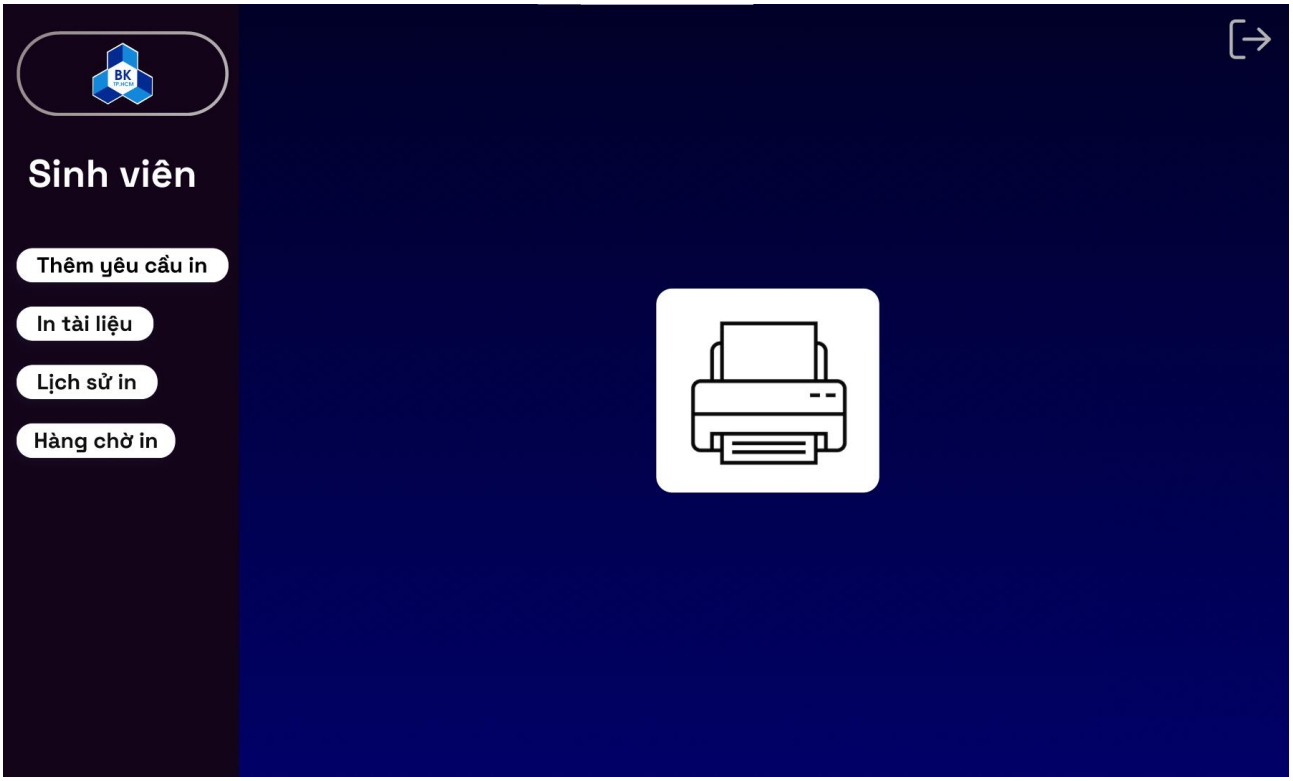
Log in screen: The log-in interface is the entry point where students and SPSO personnel authenticate via the HCMUT_SSO system to securely access the smart printing service. This ensures secure access and tailored functionality based on user roles.





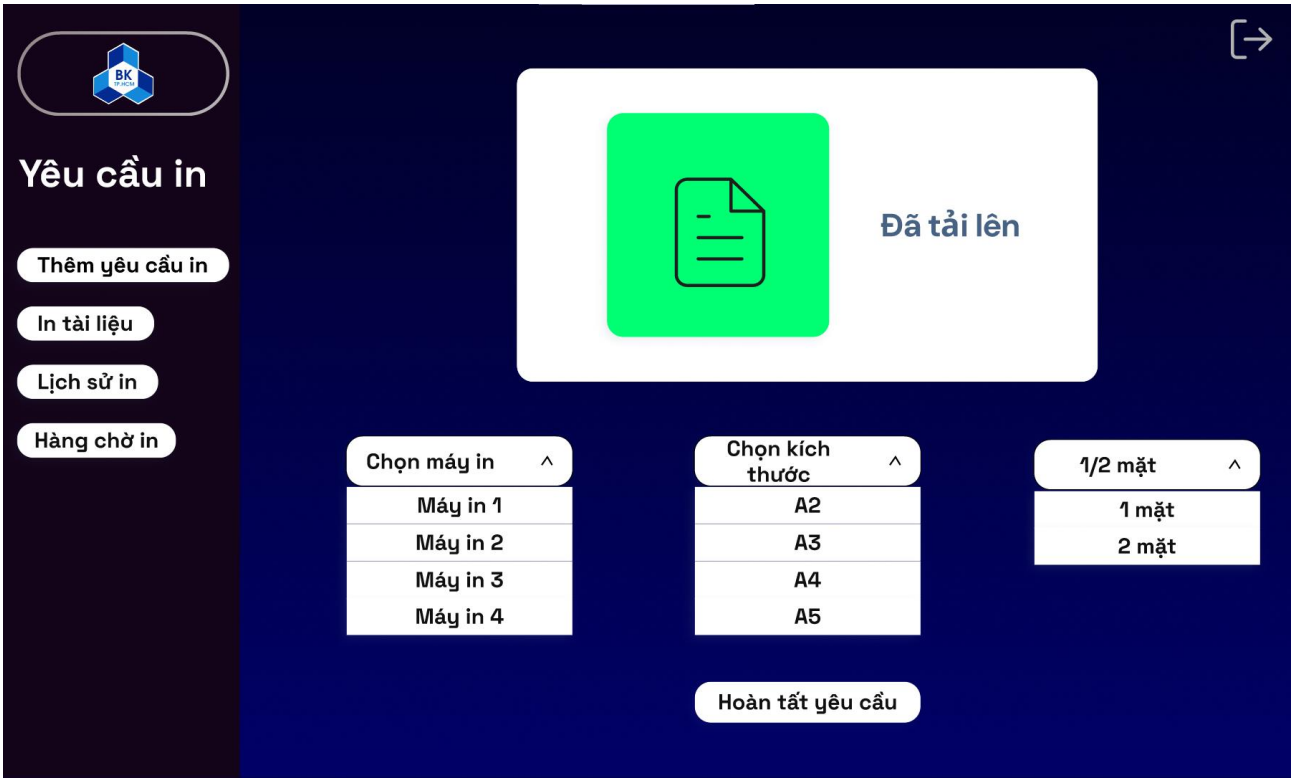
1. Student Version:

Home screen:



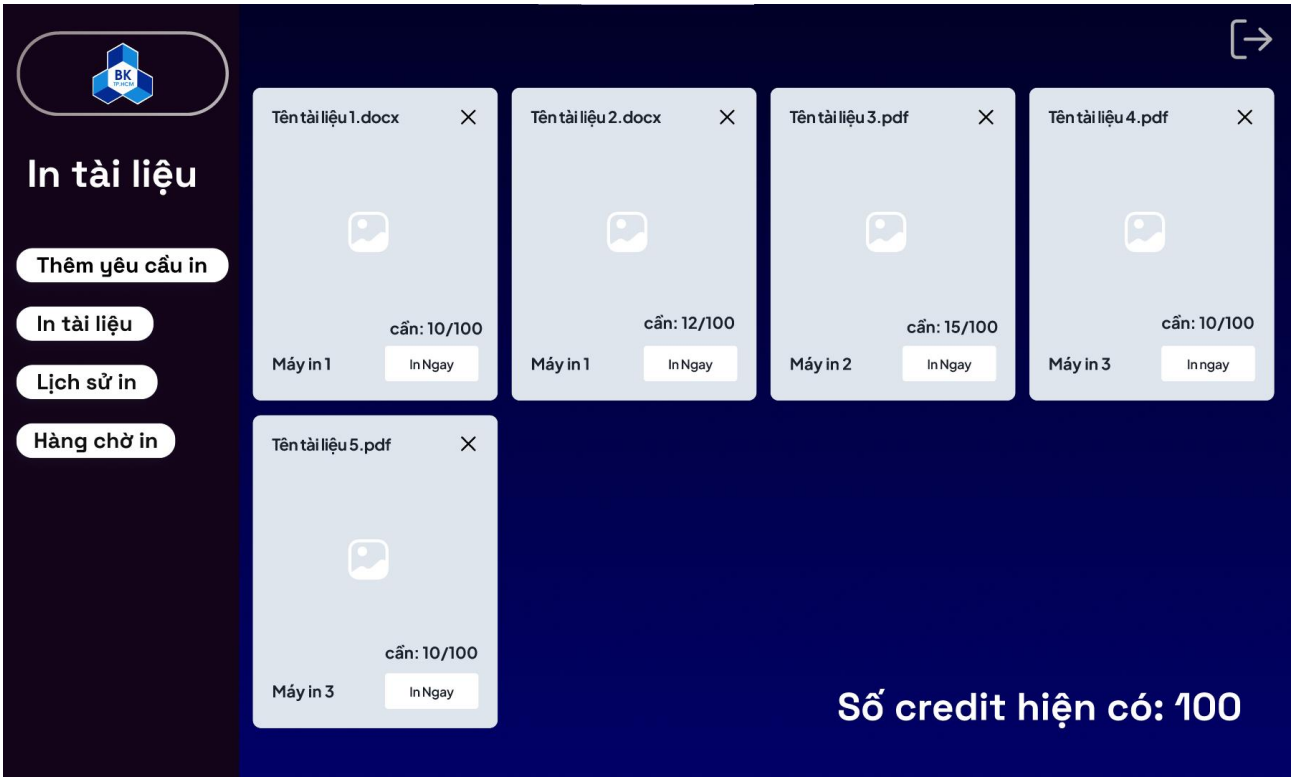
After logging in, students are presented with a user-friendly home screen that acts as the central hub, allowing them to easily manage printing requests, view history, and monitor ongoing prints.

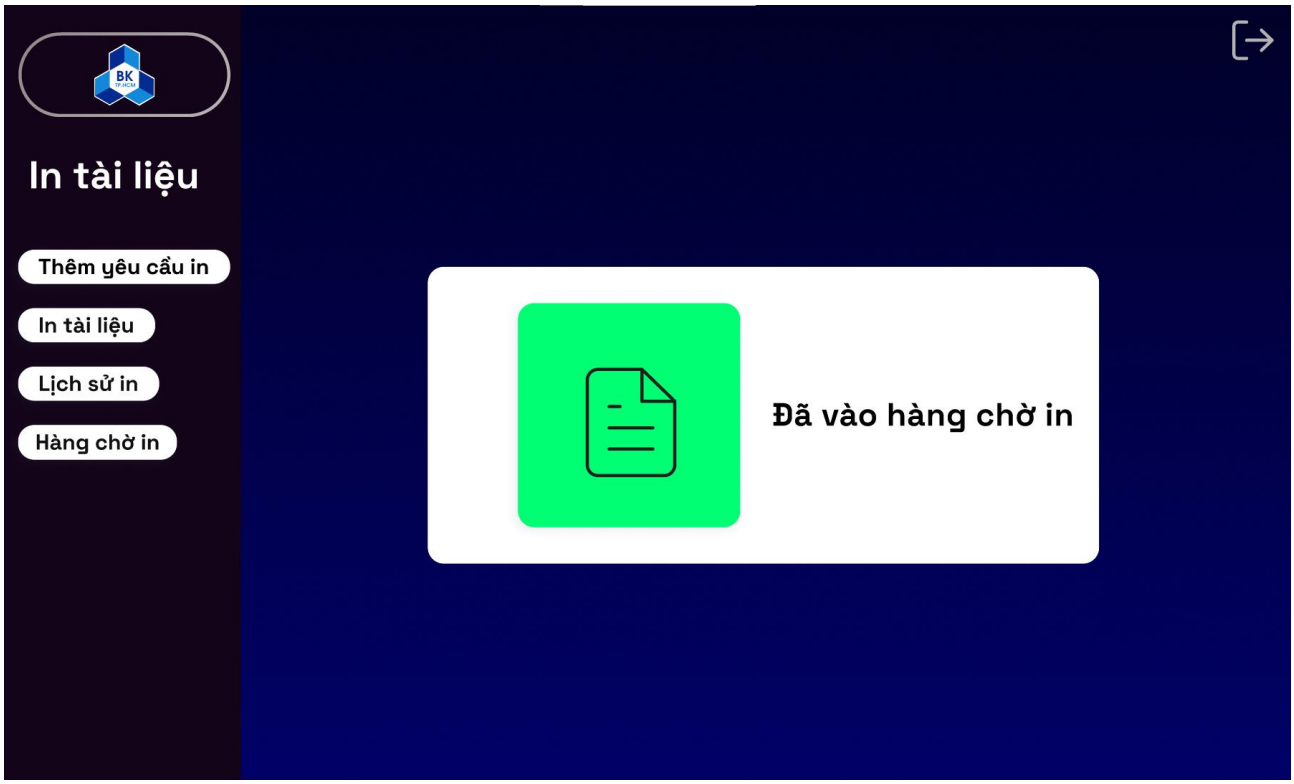
Add Printing Request:



This feature allows students to submit printing jobs by uploading documents, selecting a printer, and configuring print options like paper size, number of copies, and print quality before submitting their request.

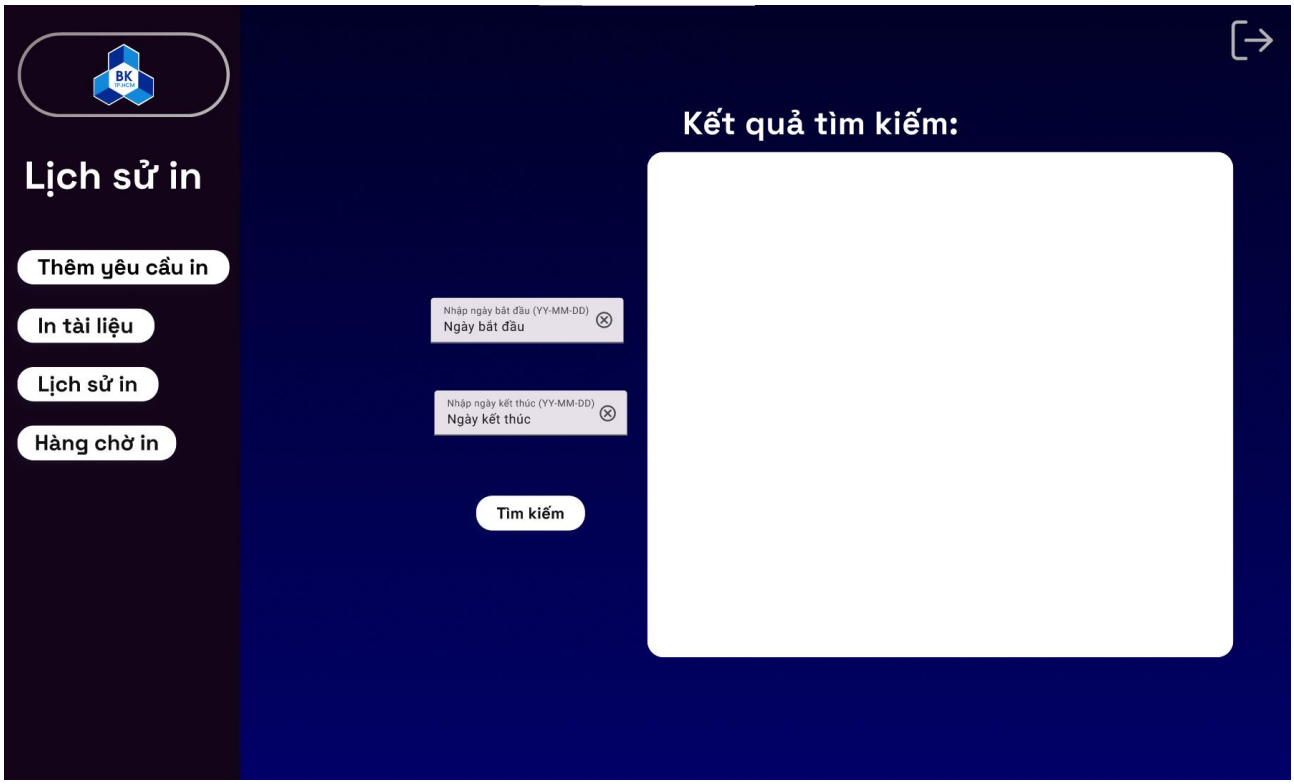
Print Document:

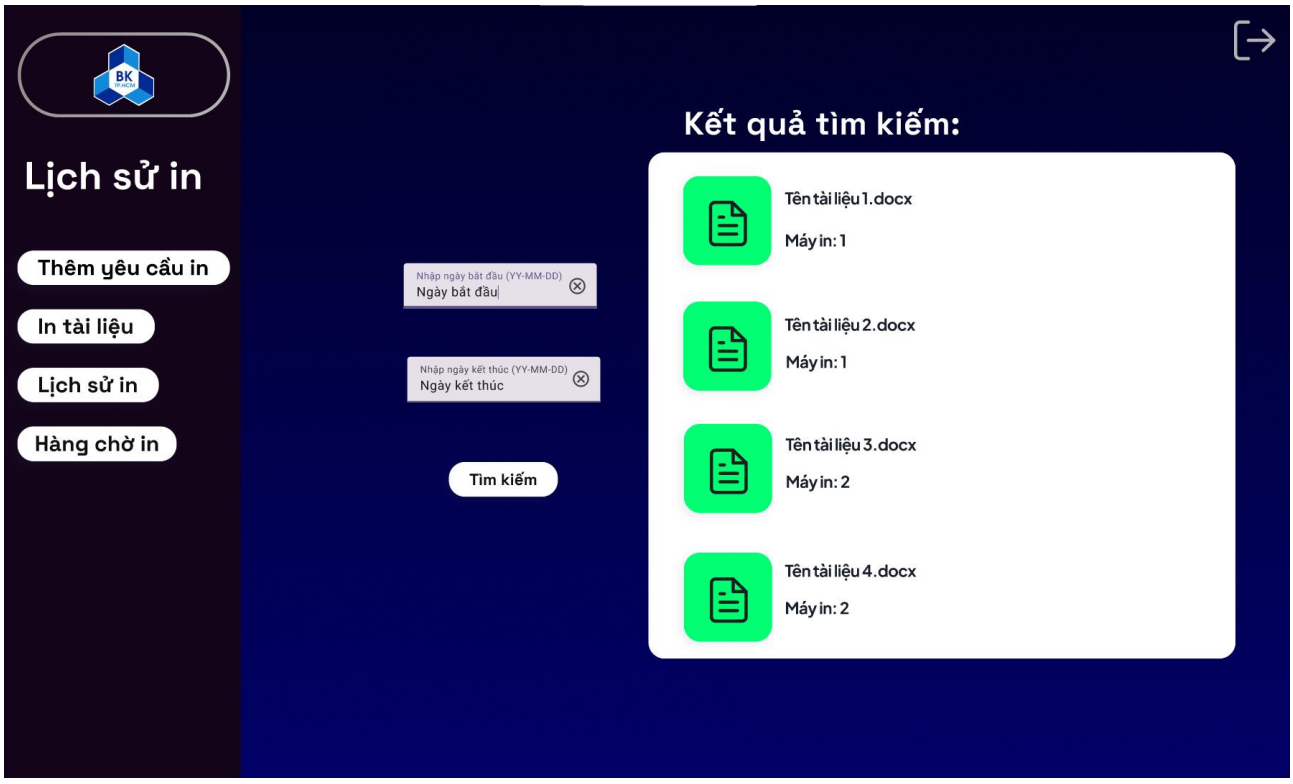




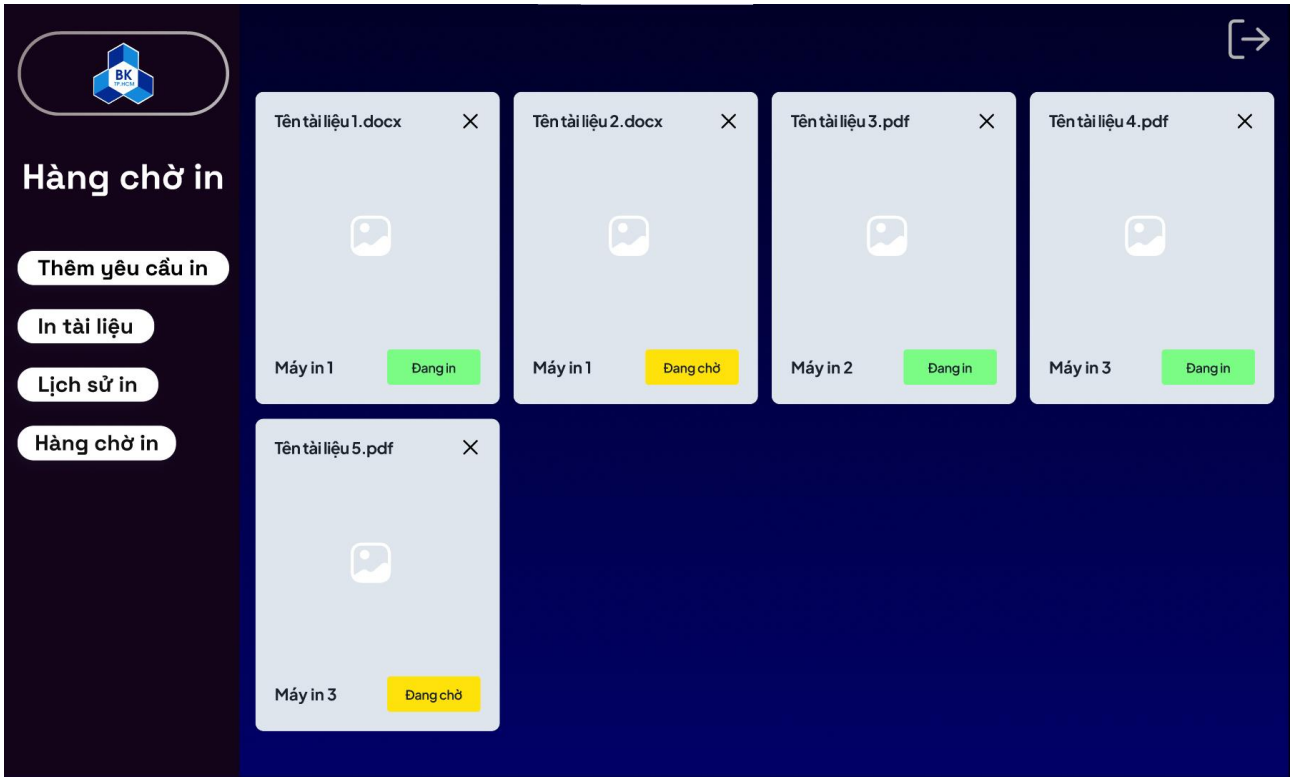
In this interface, students can confirm their uploaded document, choose a printer, and finalize the print request. The system checks for sufficient balance and available printers before processing the job.

View History: Students can view their printing history, which includes detailed records such as print dates, document names, and the number of pages printed. This feature aids in tracking usage and managing print credits.


Lịch sử in
Thêm yêu cầu in
In tài liệu
Lịch sử in
Hàng chờ in
Nhập ngày bắt đầu (YY-MM-DD)
Ngày bắt đầu
Nhập ngày kết thúc (YY-MM-DD)
Ngày kết thúc
Tìm kiếm
Kết quả tìm kiếm:


Lịch sử in
Thêm yêu cầu in
In tài liệu
Lịch sử in
Hàng chờ in
Nhập ngày bắt đầu (YY-MM-DD)
Ngày bắt đầu
Nhập ngày kết thúc (YY-MM-DD)
Ngày kết thúc
Tìm kiếm
Kết quả tìm kiếm:
Tên tài liệu 1.docx
Máy in: 1
Tên tài liệu 2.docx
Máy in: 1
Tên tài liệu 3.docx
Máy in: 2
Tên tài liệu 4.docx
Máy in: 2

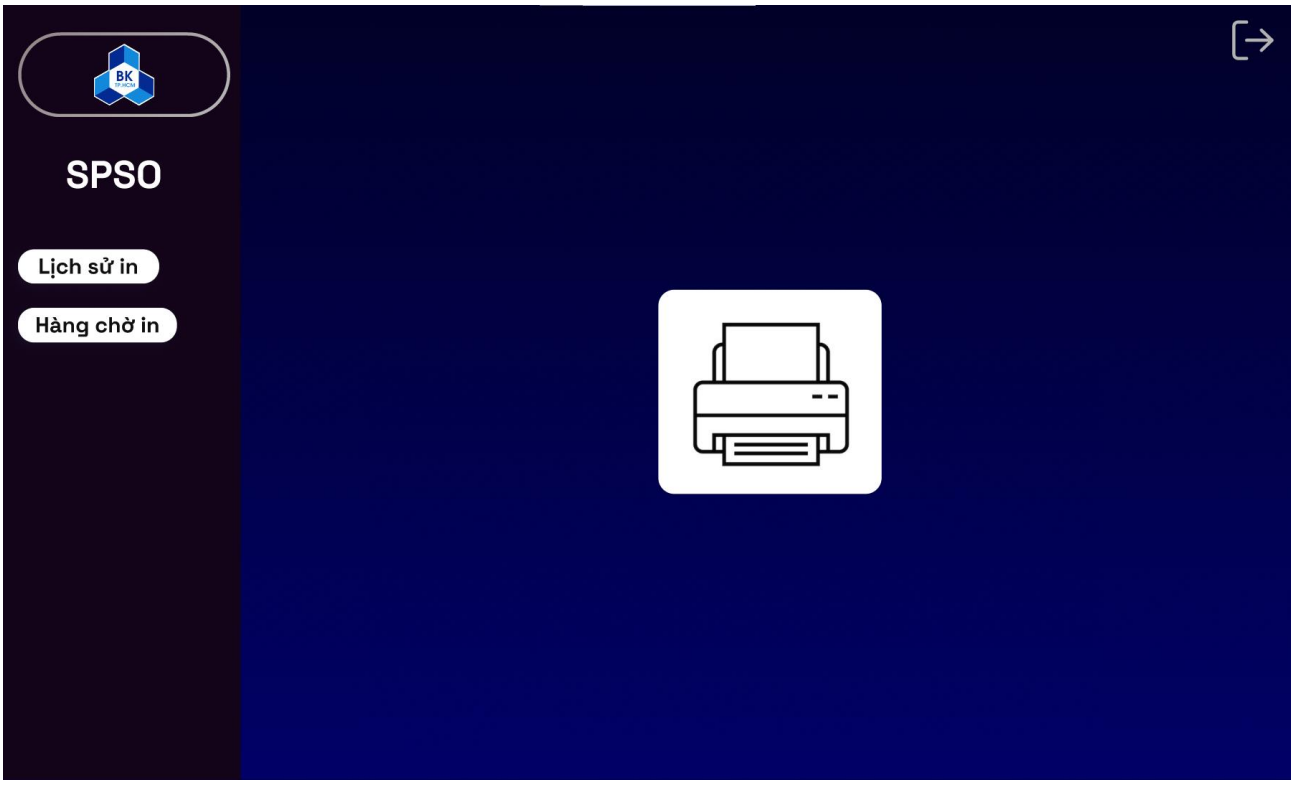
View On-going Printing Request:



This screen displays the status of all active print jobs for the student. It provides real-time updates, and students can track progress or cancel requests if necessary.

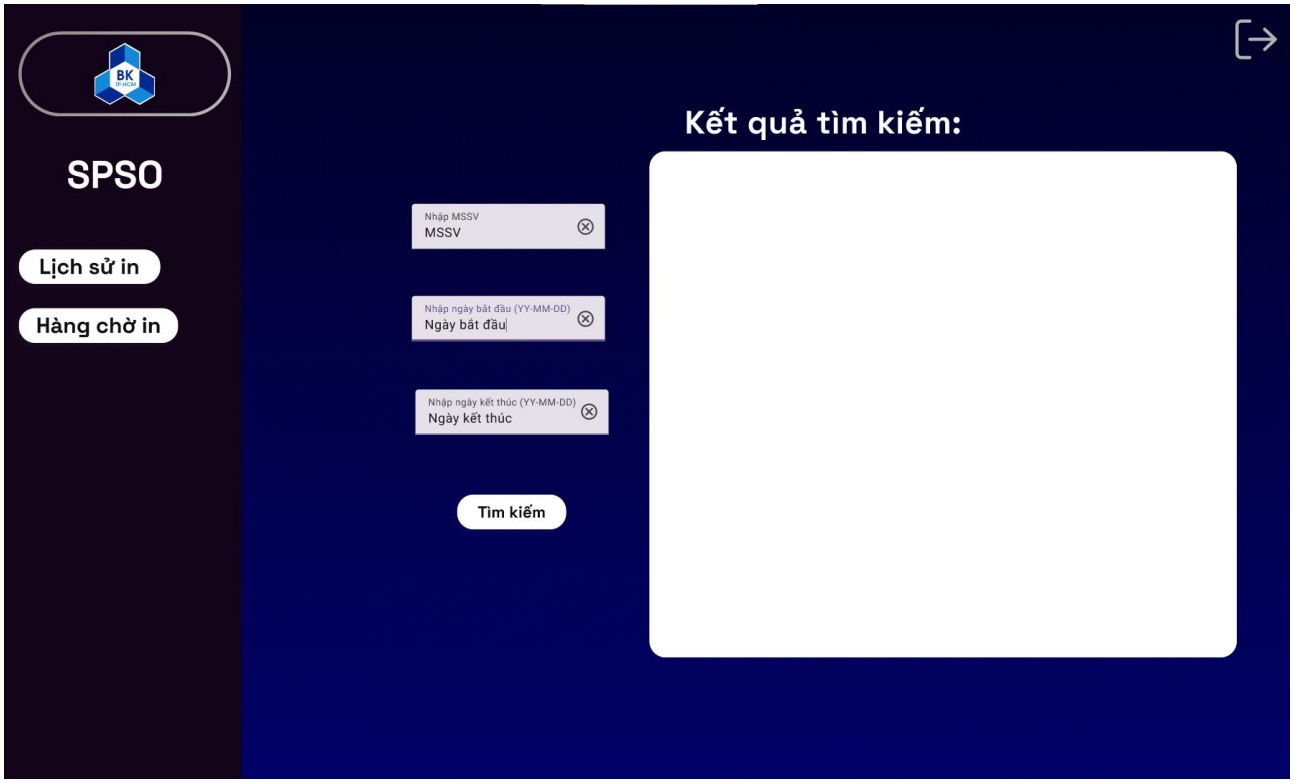
2. SPSO Version:

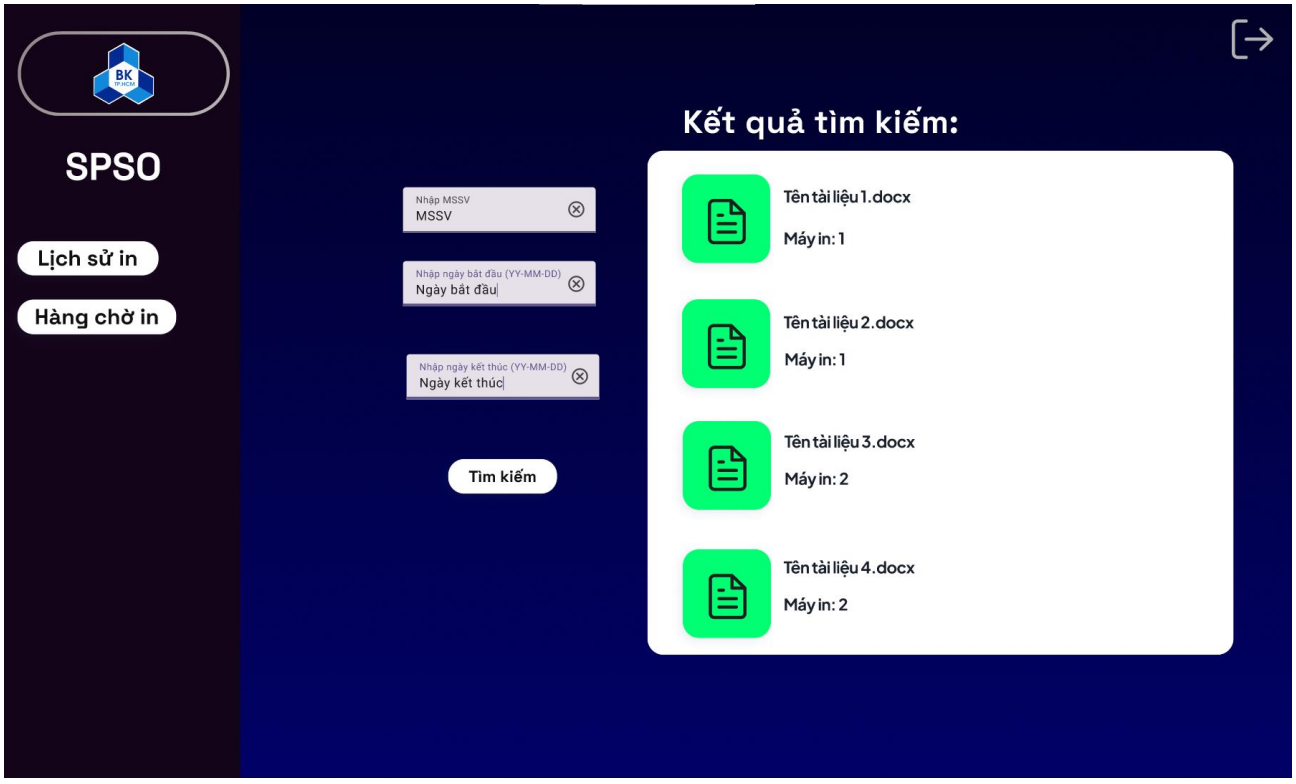
Home Screen:



The SPSO personnel have a specialized home screen that serves as a dashboard to monitor, control, and manage all active printers and student requests across the system.

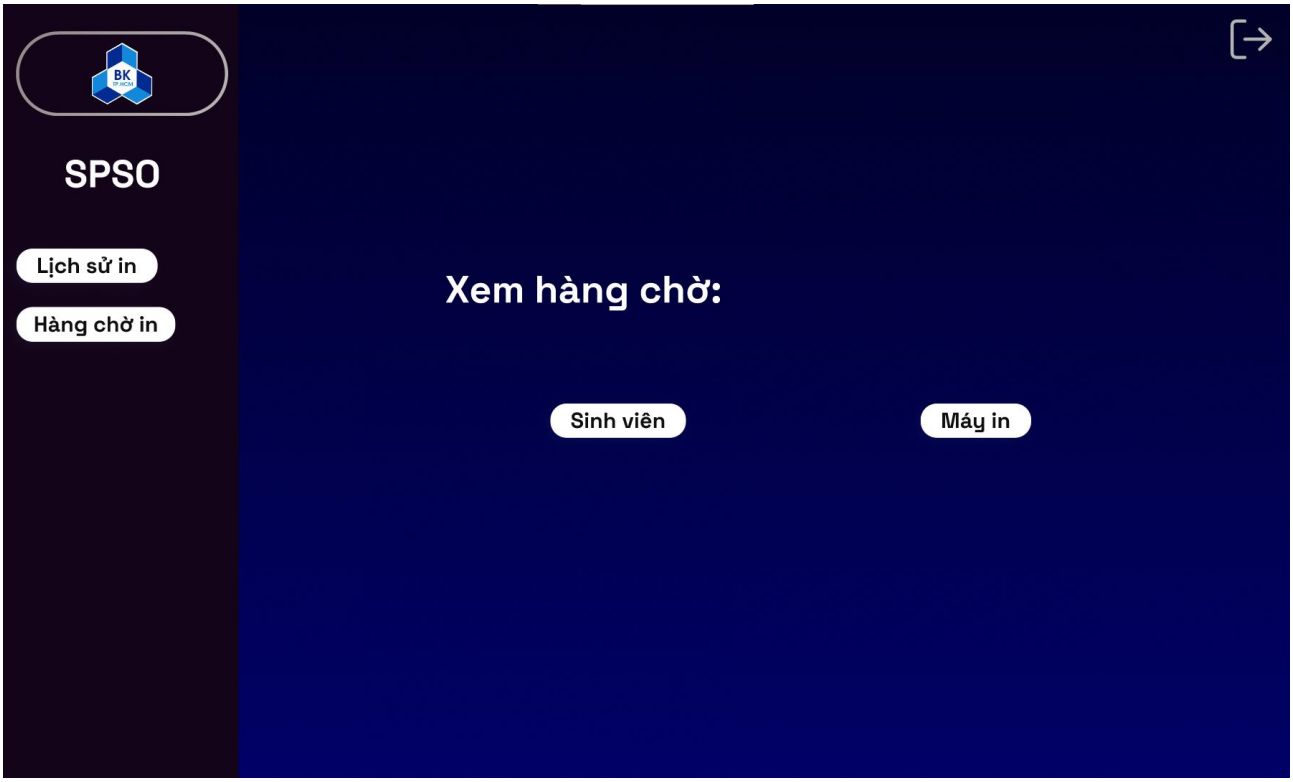
Show History:





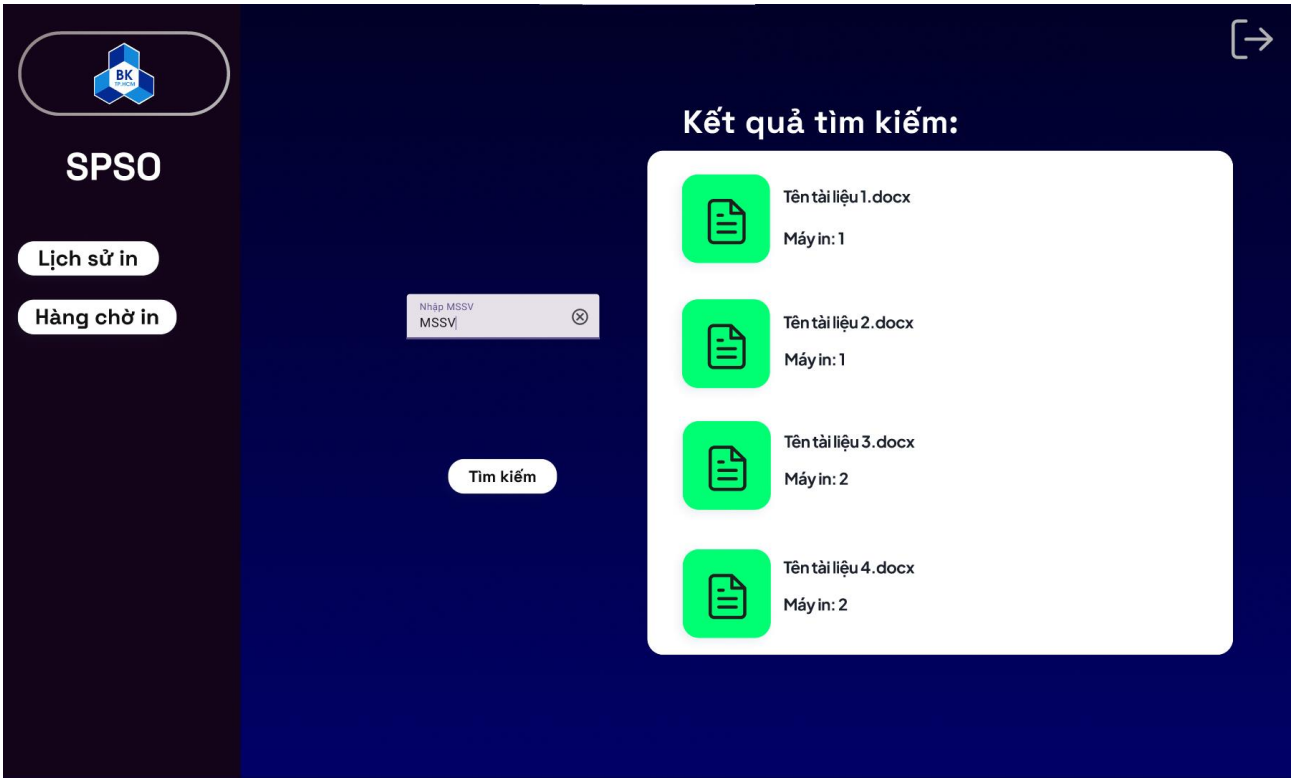
SPSO personnel can view the history of all printing activities, filterable by student or printer. This feature helps in operational management and report generation.

View On-going Printing Requests:



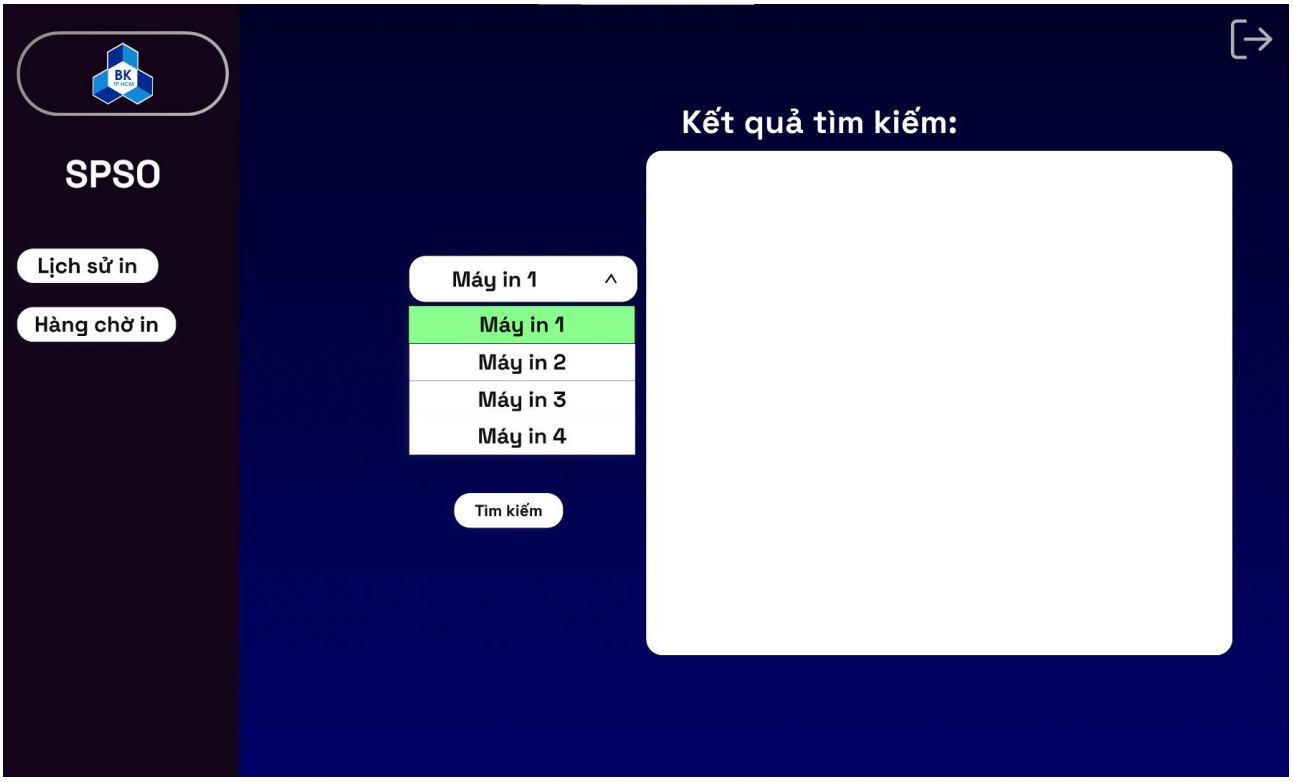
SPSO staff can monitor active print jobs across all printers or specific students. This feature allows real-time tracking of requests to ensure smooth operations.

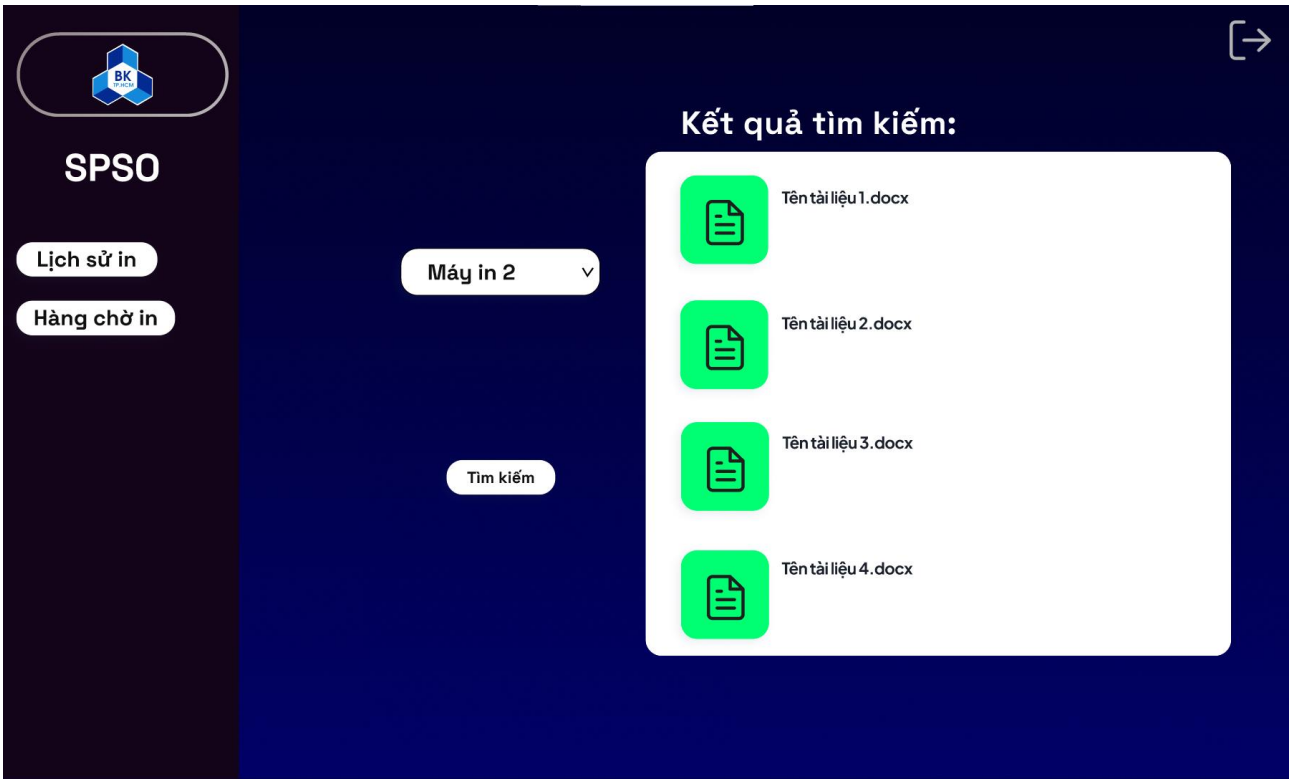
View On-going Printing Requests (Of One Student):



SPSO can narrow down ongoing requests to those of a single student, helping them manage individual student queries or troubleshoot specific cases.

View On-going Printing Requests (Of One Printer):





This feature allows the SPSO to focus on jobs being handled by a single printer, making it easier to manage printer-specific issues or workloads.

Link for demo UI of the software: https://tinyurl.com/SEAsm1DemoUI

4. Task 3: Architecture design.

4.1 Task 3.1: Architecture design based on Layered Design Pattern.

4.1.1 Introduction to Layered Design Pattern.

The *Layered Design Pattern* is an architectural approach we, as the development team, will employ to structure the Student Printing Services System (SPSS). This pattern divides the system into distinct layers, each with specific responsibilities and well-defined roles. The primary benefit of this pattern is that it enforces separation of concerns, meaning that each layer only needs to interact with the layer directly above or below it. This approach not only makes the system more modular and maintainable but also enables us to isolate issues, making testing and debugging more straightforward. By using the layered design, our team can independently develop and test each layer, improving productivity and reducing the likelihood of introducing unintended issues as the system evolves. For the SPSS, the layered pattern is particularly advantageous, as it allows us to organize complex functionalities in a manageable way and facilitates easier scaling and future enhancements.

4.1.2 Brief Analysis of Our Architecture Diagram

In the design of the SPSS, we have organized the system into four primary layers: **Presentation Layer, Business Logic Layer, Persistence Layer, and Storage Layer.**

- **Presentation Layer:** This is the top layer and represents the user interfaces. Our team has divided it into sections for different types of users. It includes a General Interface with a home page and single sign-on (SSO) login page, as well as dedicated interfaces for the SPSS Interface and the Student Interface. The SPSS Interface is designed for administrative tasks, such as managing printers and viewing usage reports, while the Student Interface allows students to print documents, purchase pages, and view their print history. This layer captures user input and displays output, serving as the primary interaction point with the system.

- **Business Logic Layer:** This layer contains the core processing logic of the SPSS and is the heart of the system. It consists of two main components: the Printing Controller and the Management Controller. The Printing Controller is responsible for managing tasks like validating student credits, handling print requests, and logging printing activity, while the Management Controller focuses on tasks such as configuring

printers, viewing activity logs, and generating usage reports. Additionally, this layer interfaces with External Services like HCMUT-SSO for authentication and BK-Pay for handling payments. By centralizing our main logic here, we ensure that the system's processing and rule enforcement are handled consistently.

- **Persistence Layer:** Acting as a bridge between the business logic and storage layers, this layer provides APIs, including the User API, Printer API, and Services API, to abstract the underlying data storage complexities. This abstraction means that the business logic layer can retrieve and store data without directly interacting with the database. This approach enables our team to update or modify the database structure without impacting other layers, which is essential for maintaining flexibility as the system grows.

- **Storage Layer:** The Storage Layer is the foundation of our design, holding the system's persistent data. Here, we store essential entities like Student, Printer, Printer’s Log, and Student's Log. This layer is accessed indirectly by the business logic layer via the persistence layer, which helps enforce data integrity and security. By isolating data storage from the rest of the system, we ensure that the underlying database can be modified, optimized, or even replaced without affecting other components.

In summary, this layered design organizes the SPSS's functionality in a clear, structured manner, allowing each layer to focus on specific responsibilities. For our development team, this approach simplifies the development process and enhances the system's flexibility. It will enable us to add new features, update components, or scale the system efficiently, with minimal impact on the overall architecture.

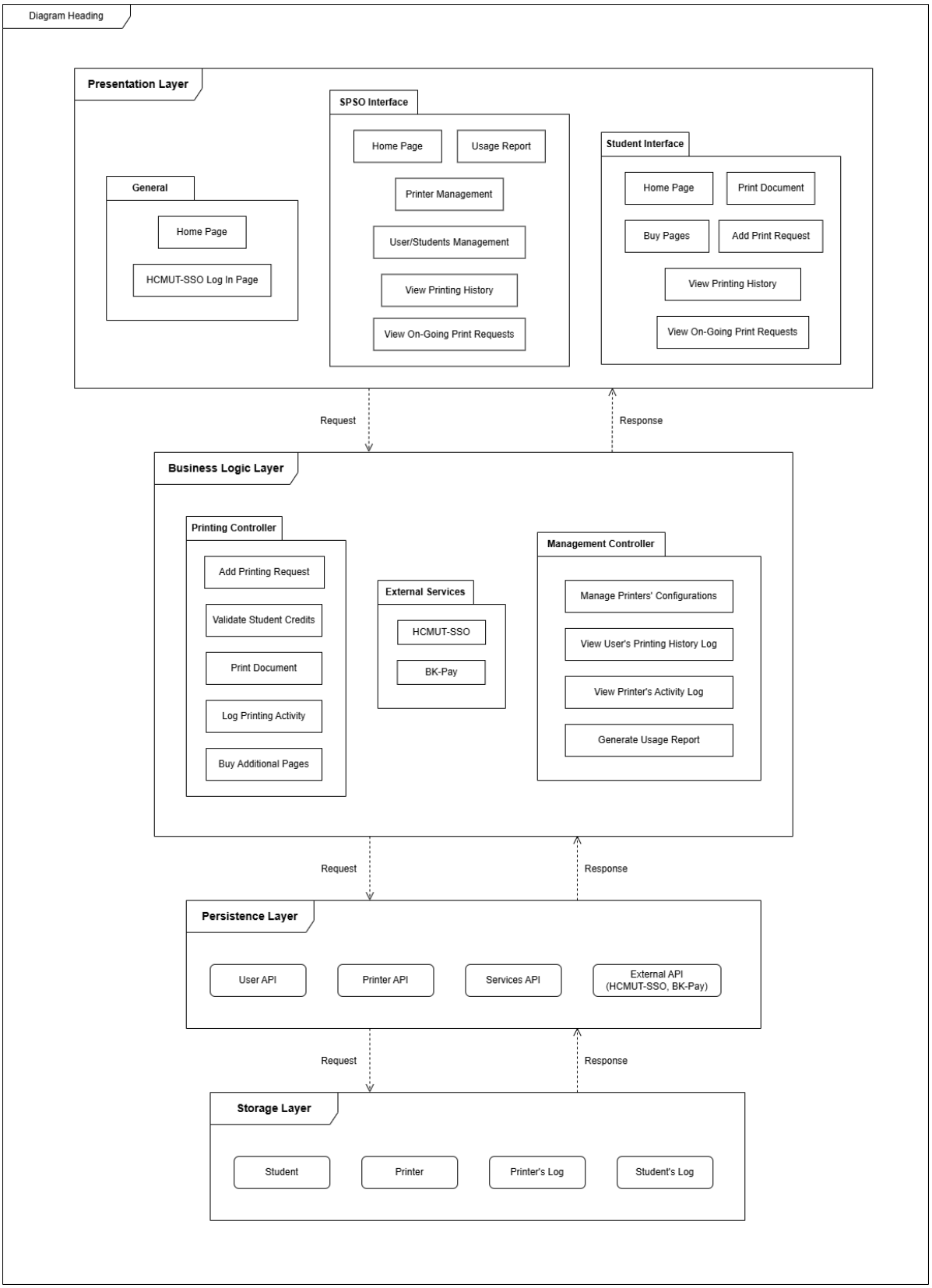


Figure: The SPSS's architecture design based on Layered Design Pattern.

4.1.3 The Presentation Layer.

The presentation Layer is the part of the system where users interact with the printing service. It is split into three main sections: General Interface, Student Interface, and Admin (SPSO) Interface.

- The General Interface includes the Home Page and Login Page using the HCMUT SSO. This is where all users start, helping them log in securely.
- The Student Interface is made for students to easily handle their printing needs. It has pages for printing documents, buying extra pages, and checking their print history. Students can also see ongoing print jobs and keep track of how many pages they've used.
- The Admin (SPSO) Interface is for staff who manage the system. It includes tools for adding and managing printers, viewing printing logs, and generating reports on printer usage. This helps administrators keep everything running smoothly.

4.1.3.1 General Interface

Home Page:

- The home page serves as the initial point of entry for all users. It provides a clean and simple layout with options to log in, view system information, and navigate to different sections depending on the user role (student or admin).
- Includes a prominent HCMUT Single Sign-On (SSO) login button for secure authentication.

Login Page:

- For both students and administrators, the login page will have options to select the user type.
- Upon successful authentication, users are redirected to their respective dashboards.

4.1.3.2 Student Interface

The Student Interface is designed to simplify printing and account management tasks for students:

Home Page:

- Display all the students' options for the admin to choose.

Add Print Request:

- Provides options to choose upload documents, choose printer, page size, and print one or both sides.

Print Document:

- Determine if there is sufficient balance for students to print the uploaded document, if sufficient, students can hit print.

Buy Pages:

- Provides a seamless interface for purchasing additional printing pages using integrated payment services like BKPay.
- Includes a simple form to input the number of pages to purchase, with real-time updates on the total cost.

View Printing History:

- Displays a detailed log of past print jobs, including file names, print dates, page counts, and printer locations.
- Includes filters to sort by date, printer, or document type for easy tracking.

View Ongoing Print Requests:

- Allows students to monitor their active print jobs, providing options to cancel or modify requests if needed.

4.1.3.3 SPSO (Admin) Interface

The SPSO interface is designed for administrators to manage the system efficiently:

Admin Home Page:

- Display all the admin's options for the admin to choose.

Printer Management:

- A comprehensive interface to add, enable, or disable printers.
- Displays printer details (brand, location, status) and allows administrators to update configurations.

User/Student Management:

- Enables viewing and managing student accounts, including updating printing quotas and resolving user issues.

View Printing History:

- Offers administrators access to a complete log of all printing activities for auditing and reporting purposes.

Usage Report:

- Provides reports and analytics on system usage, helping administrators plan resources and optimize printer allocations.

View Ongoing Print Requests:

- Allows admin to monitor all active print jobs, providing a brief view to manage printers and students.

4.1.4 The Persistence Layer (API Layer).

API management within the HCMUT_SSPS (Student Smart Printing Service) is essential for enabling secure, efficient, and scalable interactions across the system's various functions and services. The architecture includes four primary APIs: User API, Printer API, Service API, and External API, each serving distinct but interconnected roles.

4.1.4.1 3.1.3.1 User API

The User API manages user-related operations, providing secure and convenient access to the system for both students and Student Printing Service Officers (SPSOs). Authentication is handled via integration with the university's HCMUT-SSO system, allowing for single sign-on across university services and reducing the need for separate

login credentials. This API ensures that each user can only access permitted features based on their role. For students, the API facilitates profile management, viewing their printing balance, tracking quotas, and viewing their detailed print history logs. Each print log entry includes key data points like the file name, print date, and the number of pages printed, helping students monitor their usage. SPSOs access the User API to configure system settings, review students' print logs for compliance and generate customized reports for specific time periods. Data visibility is role-based, ensuring secure and appropriate data access for both students and SPSOs.

Some methods of User API:

- getUserID(string Username): Pass the Username to get the information of user ID
- getUserName(string Username): Pass the Username to get the information of user name
- getUserEmail(string Username): Pass the Username to get the information of user email
- addPrintingActivity(): Create a new printing activity and add it into the printing history list
- getPrinttingTime(string Username, int index): Pass the Username and printing activity index to get the printing time
- getPrintingFileName(string Username, int index): Pass the Username and printing activity index to get the file name
- getPrintingPageType(string Username, int index): Pass the Username and printing activity index to get the type of paper using for printing
- getPrintingPageNumber(string UserName, int index): Pass the Username and printing activity index to get the number of paper using for printing
- addTransactionActivity(): Create a new transaction activity and add it into the transaction history list
- getTransactionTime(string Username, int index): Pass the Username and transaction activity index to get the transaction time

- getTransactionPage(string UserName, int index): Pass the Username and transaction activity index to get the number of paper that student purchase
- getTransactionCost(string UserName, int index): Pass the Username and transaction activity index to get the price that students have to pay
- addReport(): Create a new report every month

4.1.4.2 Printer API

The Printer API manages all printer-related interactions, allowing SPSOs to register, configure, enable, or disable printers across the campus. Each printer's metadata includes a unique printer ID, manufacturer name, model, location details (campus, building, room number). The API ensures that printers' operational status is synchronized with the system in real time, enabling students to see only active printers available for use, minimizing downtime and errors. SPSOs can use this API to monitor printer activity and performance, facilitating efficient maintenance and troubleshooting. By allowing location-based selection, students can find and select printers near them, enhancing usability. This API also provides flexibility in adding new printers as the system scales, ensuring that the infrastructure grows with campus needs.

Some methods of the Printer API:

- getPrinterModel(string ID): Pass the printer ID to get the information of the model of printer
- getPrinterName(stringID): Pass the printer ID to get the information of printer's name
- getPrinterStatus(string ID): Pass the printer ID to get the information of printer's status
- getPrinterLocation(string ID): Pass the printer ID to get the information of printer's location
- getFilePrinted(string ID): Pass the printer ID to get the information of all files that has been printed by printer
- getPagePrinted(stringID): Pass the printer ID to get the number of paper that has been printed by printer

- addNewPrinter(): Add a new printer to the list
- changePrinterStatus(string ID): Pass the printer ID to update the status of printer

4.1.4.3 Service API

The Service API is the core of the system, orchestrating printing operations. When a student initiates a print job, this API handles file uploads, which are restricted to SPSO-configured file types for compatibility. The API validates each uploaded file and supports various print specifications, including paper size (e.g., A4, A3), page range selection, one-sided or double-sided printing, and the number of copies. Each print job is checked against the student's page balance, a feature governed by the system's quota management rules. To prevent exceeding the limit, the API ensures that students can only print up to their available balance, where one A3 page consumes the equivalent of two A4 pages. If students wish to print beyond their default quota, the API connects with the BK-Pay system (via the External API) to allow secure online purchases of additional print pages.

Some methods of Service API:

- getMaxPageNumber(): Get the maximum number of papers that students receive
- setMaxPageNumber(): Change the maximum number of papers that students receive
- checkBalance(): check the paper balance of students before printing
- connectExternalAPI(): connect to BK-Pay whenever students want to buy additional papers
- getPrintType(): Get the type that students want to print
- getMaxFileSize(): Get the maximum size of file that students can print
- setMaxFileSize(): Change the maximum size of file that students can print
- checkFileSize(): Check if the file size using for printing allow or not
- getPermittedFileType(): Get the list contains every permitted types of files to be printed

- setPermittedFileType(): Change the list contains every permitted types of files to be printed
- checkPermittedFileType():Check if the file type using for printing is allow or not

4.1.4.4 External API

The External API integrates the system with critical external services, including HCMUT-SSO for secure authentication and BK-Pay for in-system payment transactions. By using HCMUT-SSO, the system ensures single sign-on for students, creating a streamlined experience and minimizing authentication-related security risks. The BK-Pay integration allows students to buy additional print pages with online payments, which are seamlessly processed within the system. External API's robust security protocols, including encryption and token-based authentication, safeguard student data during payment transactions and authentication. Additionally, these integrations help the HCMUT_SSPS system maintain compliance with university and regulatory data privacy standards, fostering trust and security among users.

4.1.5 The Storage Layer (Data Storage Layer).

We will be using it to store our data, and our database primarily comprises three main entities: Printer, Student and Printing Log.

a. **Printer** The printer schema contains the following attributes

Attribute	Data Type	Description
Printer ID	String (Key attribute)	This represents a unique code for the printer.
Location	String	This represents the location of the specific printer.
Status	String	This indicates the printer's status as either "Available," "Busy," or "Offline."
Paper Size Available	String	This shows the number of available pages for each page size.

b. **Student** The student schema contains the following attributes

Attribute	Data Type	Description
Student ID	String (Key attribute)	This represents a unique identifier for the student.
Student Name	String	The name of the student.
Student Email	String	The email of the student.
Student Faculty	String	The faculty of the student.
Remaining Pages	Integer	The number of remaining pages for the students.

c. **Printing Log:** The printing log schema contains the following attributes

Attribute	Data Type	Description
Index	String (Key attribute)	This represents a unique index for each printing activity.
Student ID	String (Foreign key attribute)	This shows the student who submitted for the printing activity.
Printer ID	String (Foreign key attribute)	This represents the specific printer for the printing activity.
File Name	String	This shows the printed document's file name.
Printing Start Time	DateTime	This shows the start time of printing activity.
Printing End Time	DateTime	This shows the end time of printing activity.
No of Pages for Page Size	String	This represents the number of pages printed for each page size.

d. Print Job: The Print Job schema contains the following attributes

Attribute	Data Type	Description
Job ID	String (Key attribute)	This represents a unique identifier for the student.
File Name	String	This shows the document's file name submitted for printing.
Page Count for Page Size	String	This represents the number of pages printed for each page size.
Print Status	String	This represents the status of printing job as either "Queue" or "In Progress".
Submission Time	DateTime	This shows the submission time of printing job.
Printer ID	String (Foreign key attribute)	This indicates the particular printer assigned to the print job.

4.2 Task 3.2: Components diagram for Printing Services module.

This component diagram illustrates the Student Printing System (SPS), comprising four main subsystems: the Student Interface, SPS System, Database, and SPSO Staff Interface.

The **Student Interface** enables students to submit print requests, print documents, view their print history, and track ongoing requests. These requests are sent to the **SPS System**, which includes components for managing requests and coordinating printer availability to fulfill them.

The **Database** stores essential data, such as print history, student account balances, and printer request details. It supports other components by providing access to this data as needed, ensuring that transactions are recorded, and accounts are up to date.

The **SPSO Staff Interface** provides SPSO staff with the ability to view student print histories and ongoing requests, allowing them to monitor and manage the system effectively. Overall, this diagram demonstrates how the components interact to support an organized and efficient student printing service.

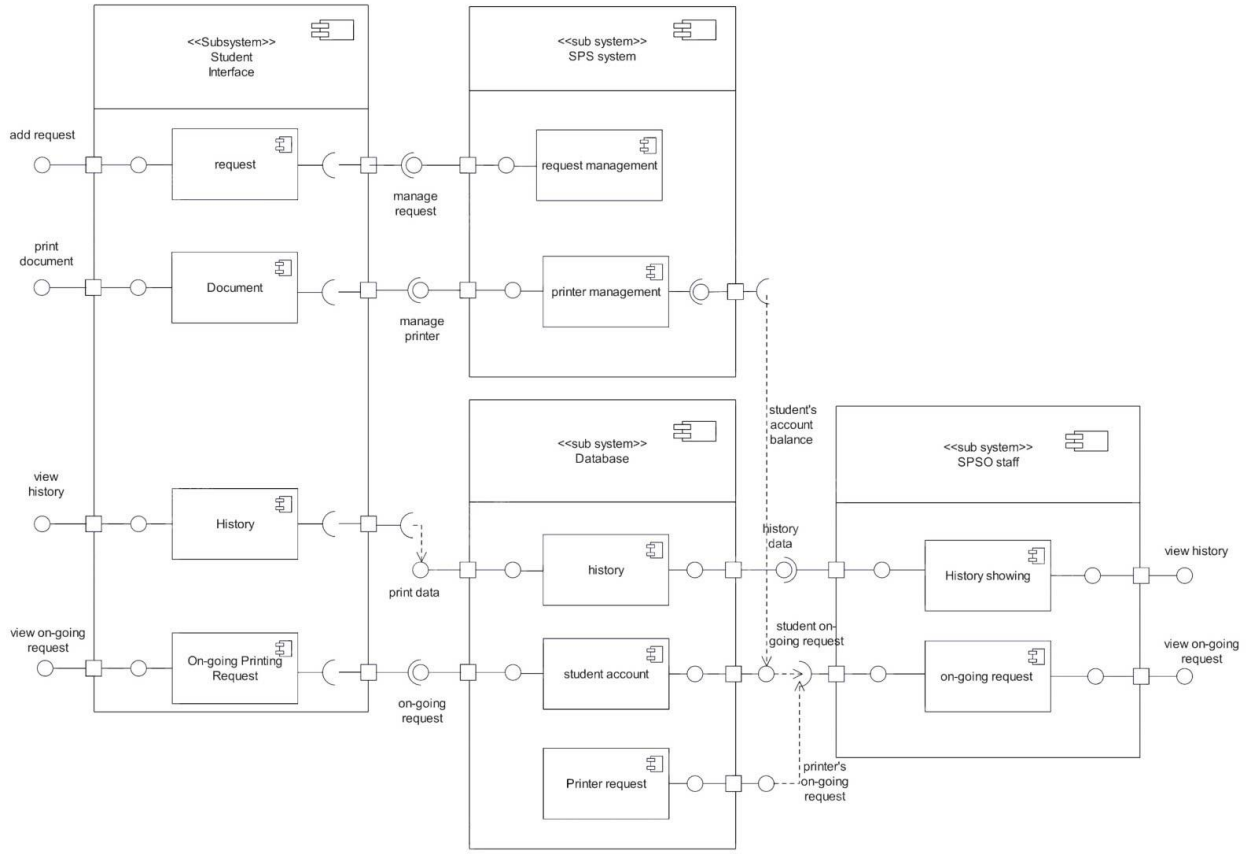


Figure: Components Diagram for Printing Services module.