
Software Requirements Specification

for

A Smart Printing Service For Students at HCMUT

Version 1.0 approved

Prepared by:

- 1. Đặng Vũ Tuấn Kiệt - 2211743**
- 2. Bùi Trọng Văn - 2213915**
- 3. Bùi Văn Quốc Bảo - 2210192**
- 4. Đặng Hoàng Khang - 2211422**
- 5. Đặng Ngọc Bảo Trâm - 2213568**

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

19/9/2024

Table of Content

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

1.1 Domain Context

1.1.1 Problem Definition

The increasing demand for printing services among Vietnamese university students, particularly at Ho Chi Minh City University of Technology, underscores the continued reliance on hard copies for academic purposes. Despite traditional print shops meeting this need, they face several limitations:

- + **Inconvenience:** Printing requests can only be made in person, requiring students to wait for their orders to be completed.
- + **File transfer issues:** Sending files to print shops via various platforms can result in errors and management difficulties.
- + **Lack of transparency:** Printing services often lack clear pricing information and invoices, making it difficult for students to track their expenses.
- + **Privacy concerns:** There is no guarantee of confidentiality when submitting documents to a print shop.
- + **Communication challenges:** Customizing print formats can be difficult due to communication barriers between students and shop owners.
- + **Long wait times:** The high number of students compared to the limited number of print shops can lead to significant delays.
- + **Accessibility:** Students often need to travel long distances to reach print shops.

1.1.2 Solutions

The HCMUT_SSPTS printing service system was developed to address the aforementioned issues. Specifically:

- + **Remote Printing:** The web-based system enables remote printing, significantly reducing wait times at physical printing stations.
- + **Student Control:** Students have full control over uploading files and customizing print settings, ensuring that their prints meet their exact specifications.
- + **Convenient Locations:** Printing facilities are strategically placed across the campus, making it easier for students to access them. Students can also choose the printing location that best suits their schedule.
- + **Credit-Based System:** The system uses the number of A4 pages as a form of currency for printing services. Students can top up their accounts and use this credit to place print requests. The cost per A4 page is fixed, and the system calculates and displays the total cost for each print job.
- + **Transaction History:** The system maintains a record of all printing and purchase history, allowing students to review their past activities.

1.2 Stakeholders and Needs

- Students at the Ho Chi Minh University of Technology (HCMUT) are the end users who want to print documents easily and quickly, with convenient payment methods.

Additionally, students need to view their document printing history and purchase print pages as needed.

- The Student Printing Service Officer manages the system and requires tools to oversee it, such as viewing individual user or printer history for easy tracking, generating statistics, and managing printer information across campus (adding, removing, or modifying printer details).
- The HCMUT, as the owner of the system, aims to provide an affordable and convenient printing service to serve students.
- The service provider aims to build and develop an intelligent printing service system that meets the users' needs and delivers it to the university.
- The online payment system seeks to integrate its payment gateway into the smart printing service, allowing students to conveniently pay for print pages.
- The printer supplier wishes to supply and integrate their printers into the student printing service system.

1.3 Benefits of the System

Benefits for Students

- **Convenience:** Place orders anytime, anywhere with an internet connection.
- **Efficiency:** Quickly select designs and customize products online.
- **Security:** Avoid in-person interactions and reduce wait times.
- **Affordability:** Benefit from competitive pricing due to efficient online processes.
- **Customization:** Easily personalize print designs with online tools.
- **Tracking:** Monitor order status and delivery schedules.
- **Variety:** Choose from a wide range of products and materials.
- **Flexibility:** Select services like express, instant, or bulk printing.
- **Payment Ease:** Use convenient payment methods like BKPay, credit cards, e-wallets, or bank transfers.
- **Sustainability:** Reduce paper usage and contribute to environmental protection.

Benefits for the University

- **Technological Advancement:** Enhance the university's tech infrastructure.
- **Smart Learning Environment:** Contribute to creating a digital-first learning space.

1.4 Functional Requirements

1.4.1 Students

1. The system shall allow students to upload document files for printing.
2. The system shall enable students to select a specific printer from available printers across campuses.
3. The system shall permit students to specify printing properties including paper size, page range, one-sided or double-sided printing, and number of copies.
4. The system shall provide students with the ability to view their personal printing history for a specified time period.

5. The system shall allow students to purchase additional printing pages using the BKPay online payment system.
6. The system shall display the current balance of printing pages for each student.
7. The system shall prevent students from printing when their page balance is insufficient for the requested print job.

1.4.2 Students Printing Service Officer (SSPO)

1. The system shall allow the SPSO to view the printing history of all students or a specific student for a defined time period.
2. The system shall enable the SPSO to filter printing history by specific printers or groups of printers.
3. The system shall provide the SPSO with the ability to add, enable, or disable printers in the system.
4. The system shall allow the SPSO to configure system parameters including the default number of pages per student, dates for allocating default pages, and permitted file types.
5. The system shall generate and store monthly and yearly reports of system usage automatically.
6. The system shall allow the SPSO to view stored usage reports at any time.

1.4.3 HCMUT Administration

1. The system shall authenticate all users through the HCMUT_SSO authentication service before granting access.
2. The system shall allocate a default number of A4-size pages to each student at the beginning of each semester.
3. The system shall maintain separate printing balances for each student.
4. The system shall log all printing actions, including student ID, printer ID, file name, printing start and end time, and number of pages for each page size.
5. The system shall provide access to the printing service through both a web-based application and a mobile application.

1.4.4 Online Payment System Provider

1. The system shall integrate with the BKPay online payment system for processing student payments.
2. The system shall record all payment transactions, including the amount paid and the number of pages purchased.

3. The system shall immediately update a student's page balance upon successful payment processing.
4. The system shall provide a secure interface for transmitting payment information.
5. The system shall provide the provider with the ability to view the complete payment history of all students, including the ability to filter transactions by date range.

1.4.5 Service Provider

1. The system shall support integration with multiple printer brands and models.
2. The system shall provide a mechanism for adding new file types to the list of permitted file types.
3. The system shall maintain a record of all system configuration changes.
4. The system shall provide error logging and reporting capabilities for troubleshooting.
5. The system shall allow for the configuration of printer-specific properties such as location and capabilities.

1.5 Non- Functional Requirements

1.5.1 Performance

- The system can handle multiple print requests concurrently, up to 1000 requests.
- The system must be scalable to support up to 1000 concurrent access requests while maintaining optimal performance.
- The response time for a print request should not exceed 5 seconds.

1.5.2 Availability

- The service should be available most of the time, at least 95% during the university schedule.
- The recovering time when the system suddenly crash should be at most 15 minutes.
- Time to maintain must be scheduled and announced ahead, and should only takes up to 1-2 hours.

1.5.3 Adaptability

- Must support all devices(laptop or phone). Website should be responsive.
- The site can run on most of the browser's versions such as Chrome 118, Firefox 118, Edge 117, Safari 17, Opera 103
- Must support service for different printing machines, type of printing.

1.5.4 Security

- All user must be authenticated via HCMUT_SSO authenticator before using the service.
- Payment service must follow PCI DSS.
- User's information must be secret only to the user.
- Protective against different type of cyper attacks.

1.5.5 User Experience

- The printing service should run swiftly, minimal delay
- Tools to keep track of the printing process.

1.5.6 User Interface

- Website must be comprehensive, easy to follow
- Must be responsive
- Support languages.

2. Use-case Diagrams (1.3)

2.1 Use-case Diagram for the Whole System

2.1.1 Actors

Actor	Description
Student	Main customer for the printing services.
SPSO	System manager.
Payment System	The system to help students buy more pages.

2.1.2 Use-case Diagram

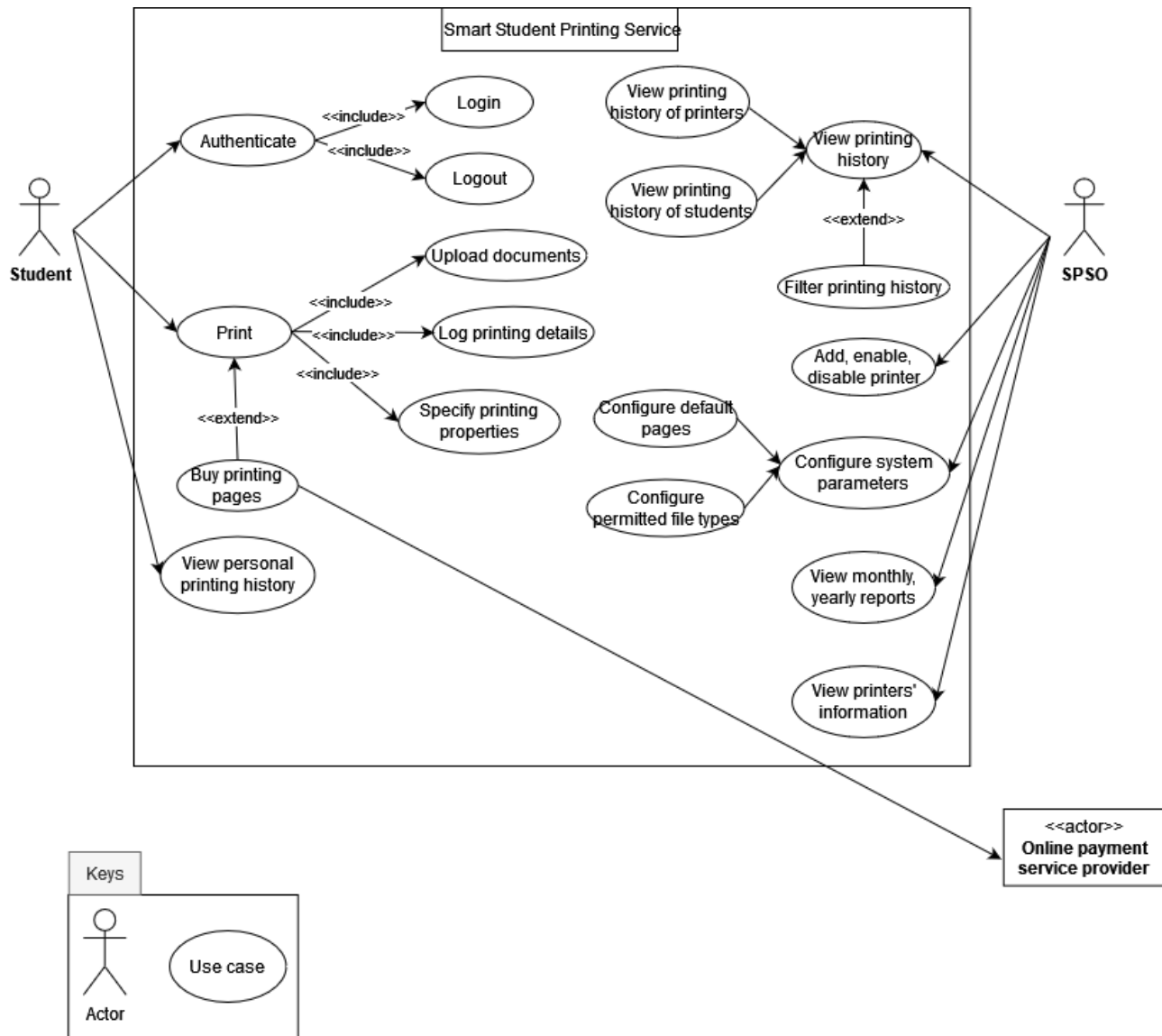


Fig 2.1 Use-case diagram for the whole system

2.2 Use-case Diagram for Printing Modules

2.2.1 Use-case of Printing Modules

Name	Description
Upload documents	Students upload the documents need to be printed.
Specify printing properties	Students choose some printing configuration.
Printing log	The system send and save the printing results.
Buy Printing Pages	Students buy more page for printing.

2.2.2 Use-case Diagram

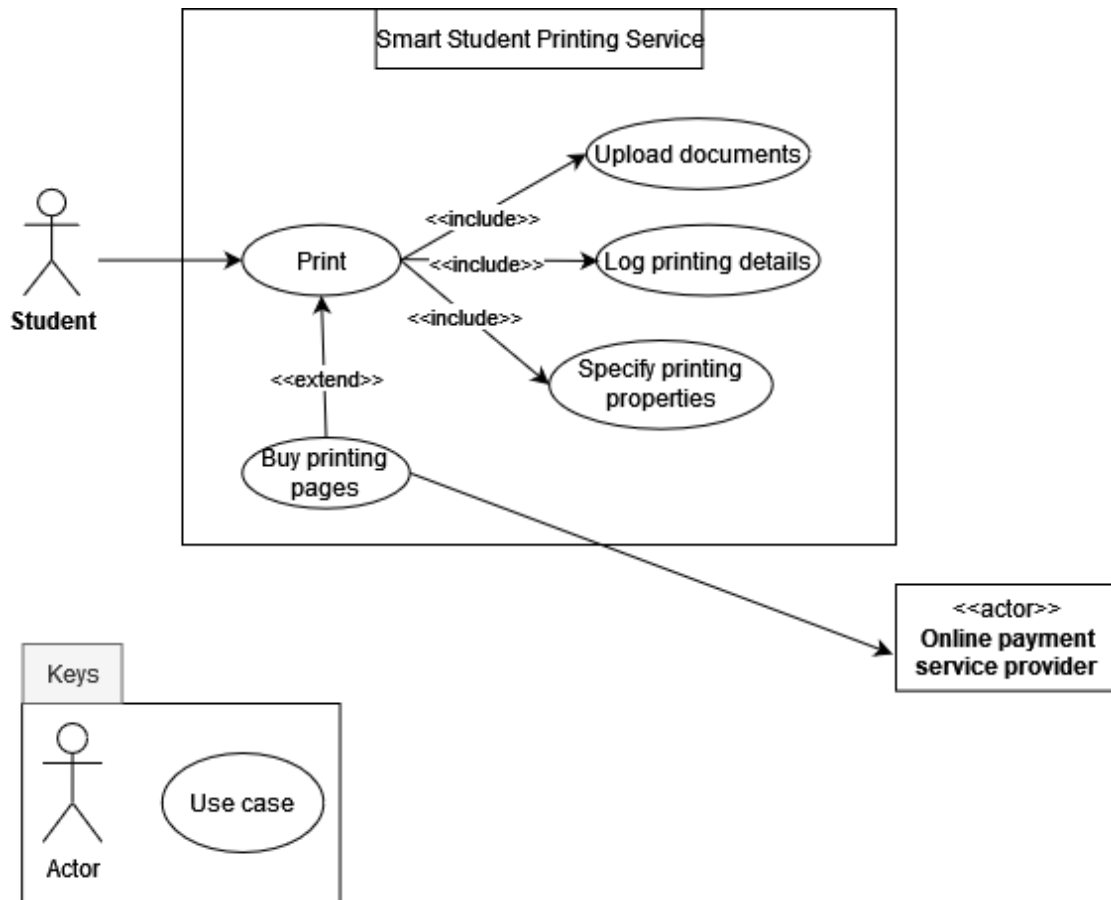


Fig 2.2 Use-case diagram for Printing Module

2.3 The Details of Use-case in Printing Module

2.3.1 Use-case Upload documents

Use Case Name	Upload Documents
Create By	Bùi Trọng Văn
Date Created	28/09/2024
Last Updated By	Bùi Trọng Văn
Date Last Updated	28/09/2024
Actors	Student
Description	Student upload file to request printing service
Trigger	Student need to print their file

Pre-conditions	Student has log in Student access to the printing request
Post-conditions	The document has been uploaded to the system.
Normal Flow	1. Student log in to the application 2. Student access to the printing request 3. Student upload file to print 4. Printing interaction is rendered
Alternative Flows	1. Student request on-capture printing 2. Student capture and upload the image 3. Printing interaction is rendered
Exception Flows	File uploads is not supported File not found
Notes and Issues	

2.3.2 Use-case Specify printing properties

Use Case Name	Specify Printing Properties
Created By	Đặng Vũ Tuấn Kiệt
Date Created	26/9/2024
Last Updated By Dated Last Updated	Đặng Vũ Tuấn Kiệt 28/9/2024
Actors	Students, Printers
Description	After uploading the file, the students can adjust some printing configuration.
Trigger	Students need to choose some printing configuration.
Pre-conditions	<ul style="list-style-type: none"> The students need to log in to the systems. Student's devices has internet connection. Student has uploaded the document to the system.
Post-conditions	The printing configuration is sent to the printers.
Normal Flow	1. Students choose the paper size, pages (of the file) to be printed, printing type: one or double sided and number of copies. 2. Students choose the printers. 3. Students confirm the printing configuration and send to the printers.
Alternative Flows	3a. Students don't confirm the printing configuration and continue to adjust. 3b. Students confirm and save the configuration for the next printing times.
Exception	
Note and Issues	

2.3.3 Use-case Printing Log

Use Case Name	Printing Log
Created By	Đặng Hoàng Khang
Date Created	28/9/2024
Last Updated By Dated Last Updated	Đặng Hoàng Khang 28/9/2024
Actors	Students, SPSO, Printer
Description	After uploading the file, the students can adjust some printing configuration.
Trigger	Students click print document.
Pre-conditions	<ul style="list-style-type: none"> • The students need to log in to the systems. • Student's devices has internet connection. • Student has uploaded the document to the system. • Student configured their printing.
Post-conditions	Students's printing request processed successfully; Recording their activity in the system.
Normal Flow	<ol style="list-style-type: none"> 1. Students select printer which be available. 2. Students select date and time to pick up printing document or auto. 3. Students confirm request and click send. 4. SPSO sends printing request to the system for selecting printer. 5. After printing process is complete, printer sends completion notification to students. 6. System saves information about printing activity to printing history, including student ID, printer ID, file name, start and end time, along with number of printed pages corresponding to each page size.
Alternative Flows	<ol style="list-style-type: none"> 2a. Students can select time in the future if needed. 6a. Students can view printing history.
Exception	<ol style="list-style-type: none"> 3a. If the total number of pages to be printed exceeds the number of pages remaining in student's account, the system will notify that there is a shortage of printed pages and request to purchase additional printed pages or edit the print parameters. 5a. If an error occurs during the printing process of the printer and the system receives an error signal from the printer, the system will send a notification to the student and the Admin.
Note and Issues	

2.3.4 Use-case Buy Printing Pages

Use Case Name	Buy Printing Pages
Created By	Đặng Vũ Tuấn Kiệt
Date Created	28/9/2024
Last Updated By Dated Last Updated	Đặng Vũ Tuấn Kiệt 28/9/2024
Actors	Students, Payment System
Description	Students buy new printing page through payment system.
Trigger	Students need more page to print their document.
Pre-conditions	<ul style="list-style-type: none"> • The students need to log in to the systems. • Student's devices has internet connection.

	<ul style="list-style-type: none"> • Student has uploaded the document to the system. • Student configured their printing. • Student confirm and request the printer to start printing.
Post-conditions	<p>The request to purchase new page has been successfully completed.</p> <p>The number of printed pages in the student's account has been updated.</p>
Normal Flow	<ol style="list-style-type: none"> 1. Students access the system's printing credit purchasing function. 2. Students enter the number of pages they wish to purchase. 3. Students select a payment method. 4. The payment service confirms the successful transaction. 5. The system records the payment history. 6. The system updates the student's account balance.
Alternative Flows	
Exception	<p>If a student tries to purchase a negative number of pages, an error will occur.</p> <p>4a. If an error occurs during the payment process, the system will display a payment failure message.</p>
Note and Issues	

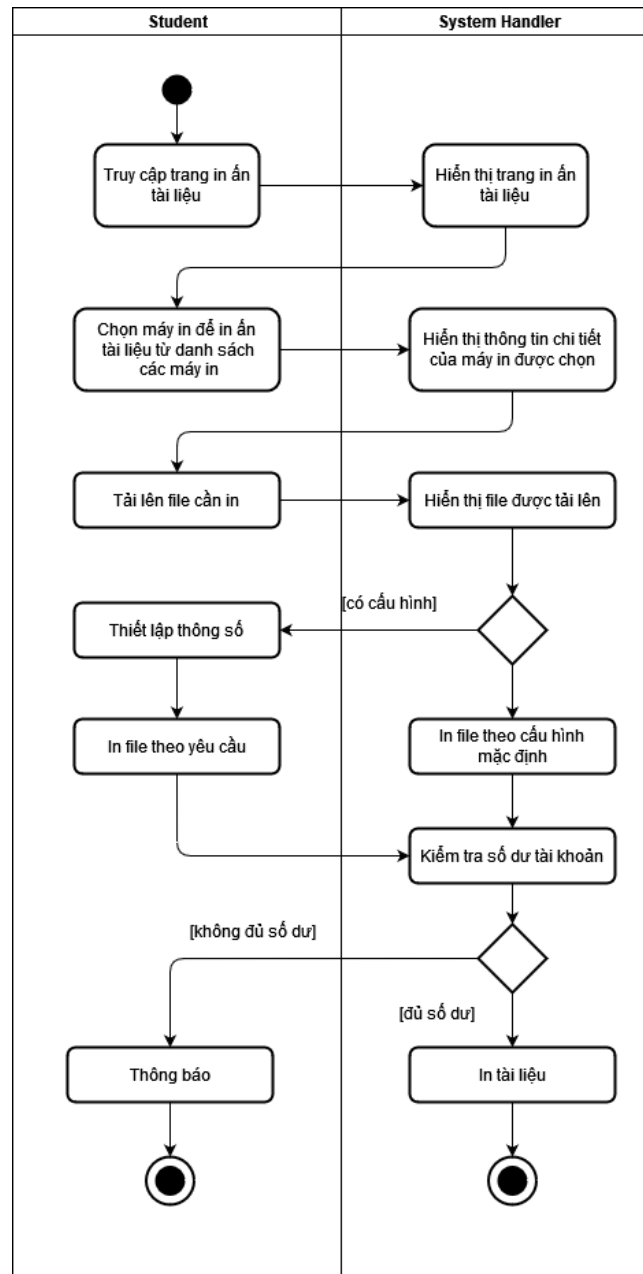
3. System Modeling (Task 2)

3.1 Activity Diagram

3.1.1 Print

Description:

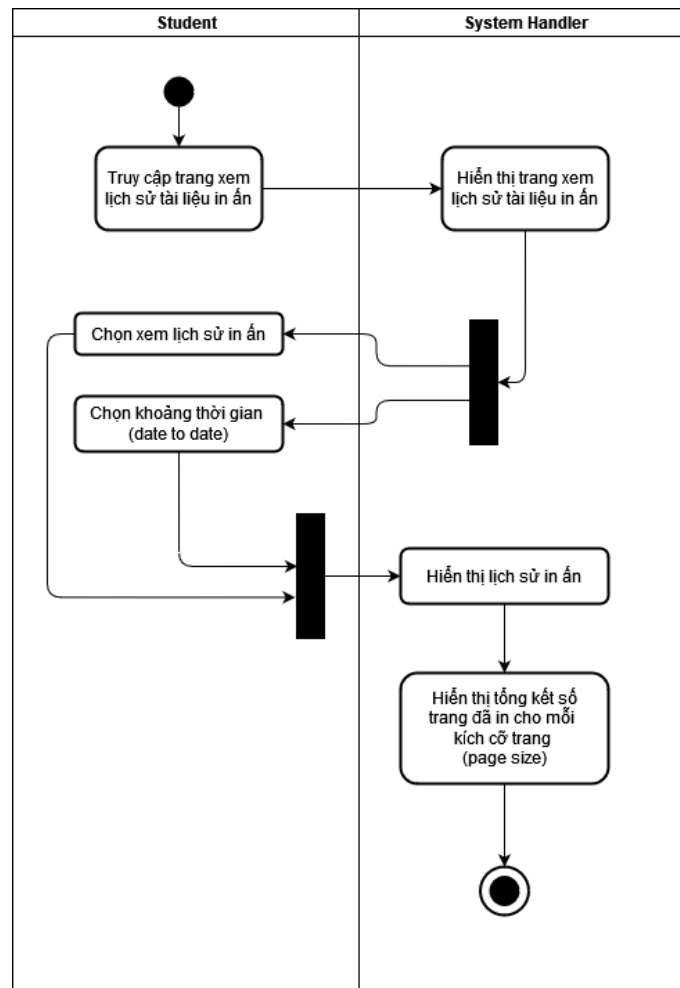
Direct Document Printing. The user accesses the "Document Printing" page, and the system displays the page screen to the user. The user selects a printer from the list to print documents; the system displays detailed information about the selected printer. The user uploads the file to be printed, and the system shows a preview of the uploaded file along with corresponding input fields to enter print configuration information. If the user inputs configuration details, the system saves the user's selections and prints according to that configuration if printing is permitted. If no configuration is entered, and if printing is permitted, the system uses the default preset configuration. Next, the system checks the account balance; if there are sufficient funds, it proceeds to print the document. If not, it notifies the user of insufficient balance and redirects them to the "Purchase Additional Print Pages" page.



3.1.2 View personal printing history

Description:

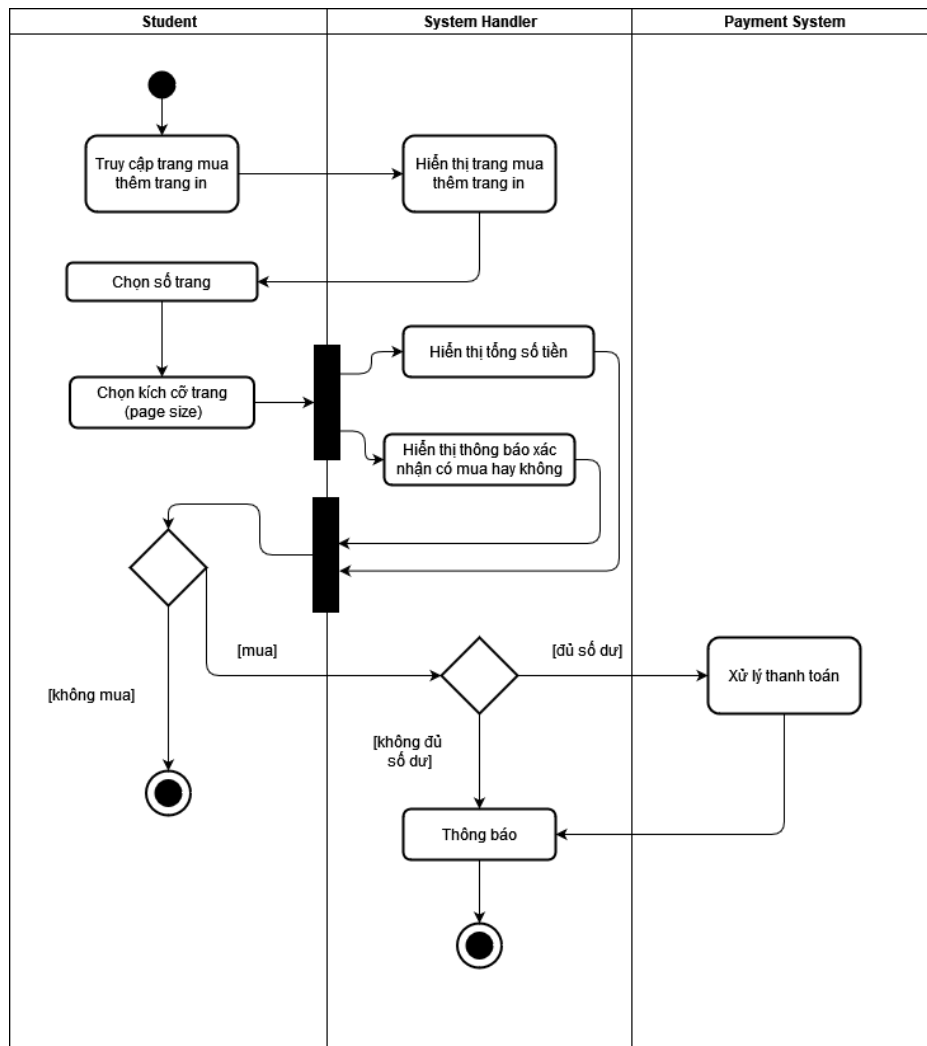
Review the user's print history within a selected time frame, along with a summary of the number of pages printed for each page size. The user accesses the "View Printed Document History" page, and the system displays the page screen to the user. The user begins parallel actions: choosing to view the print history and selecting the time period (date to date). The system exports print log data for the specified dates and calculates the number of pages printed for each page size (such as A4, A5, etc.).



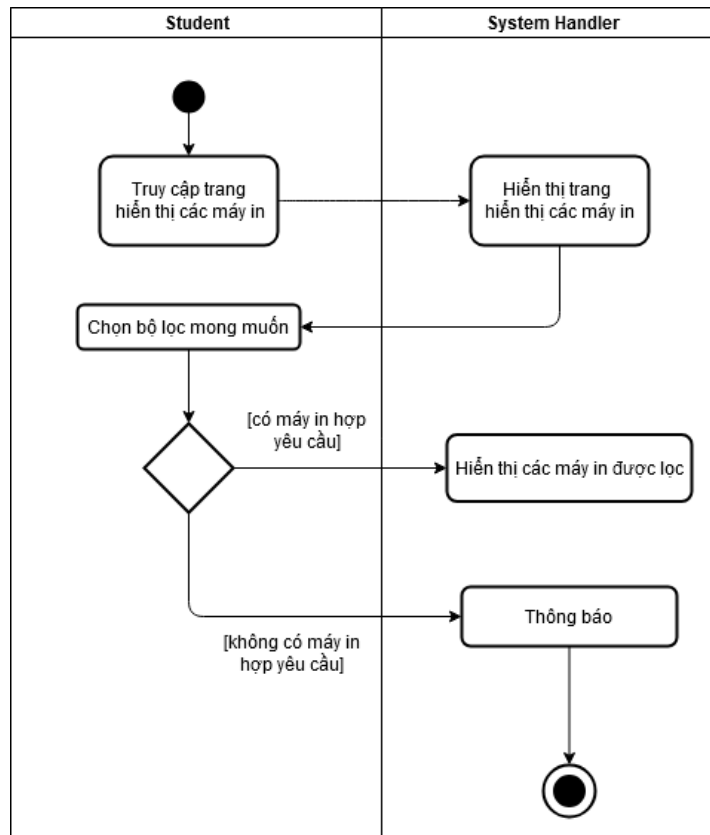
3.1.3 Buy printing pages

Description:

Description: Purchase additional print pages in the printing system via the BK Pay online payment system. The user accesses the "Purchase Additional Print Pages" page, and the system displays the page screen to the user. The user then enters the desired number of pages and the page size they wish to buy. The system concurrently displays the total amount and a confirmation message asking whether the user wants to proceed with the purchase. If the user confirms the purchase, the system checks the user's balance. If there are sufficient funds, BK Pay processes the payment, notifies the user, and concludes the transaction. If the balance is insufficient, the system informs the user and ends the process. If the user decides not to purchase, the process concludes.



3.1.4 View printers' information

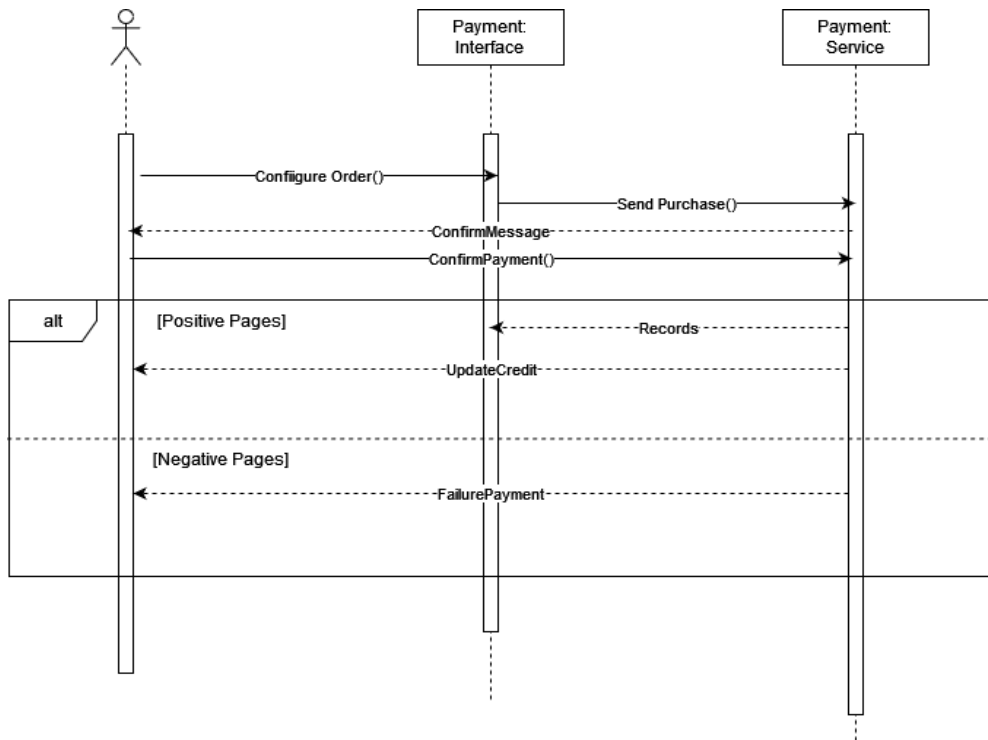


Description:

Check the status of printers in the system by filtering machines according to specific requirements. The user accesses the "Display Printers" page, and the system displays the page screen to the user. The user uses the status filter to select the statuses they want to filter by. If there are printers matching the filter criteria, the system returns the filtered printers to the user. If no suitable printers are found, the system displays a message informing the user that no printers match the criteria and concludes the process.

3.2 Sequence Diagram

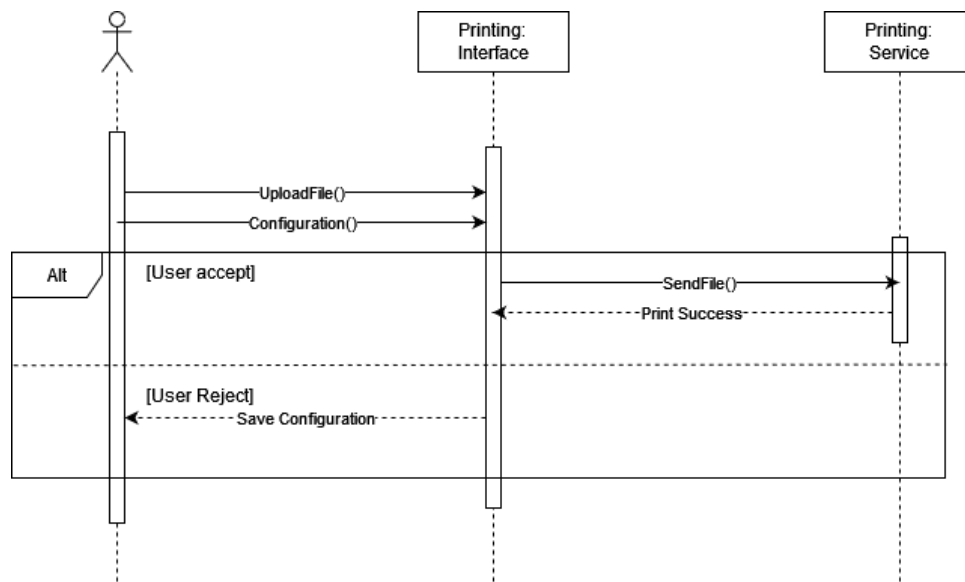
3.2.1 Diagram of “Buy Printing Pages”



Description:

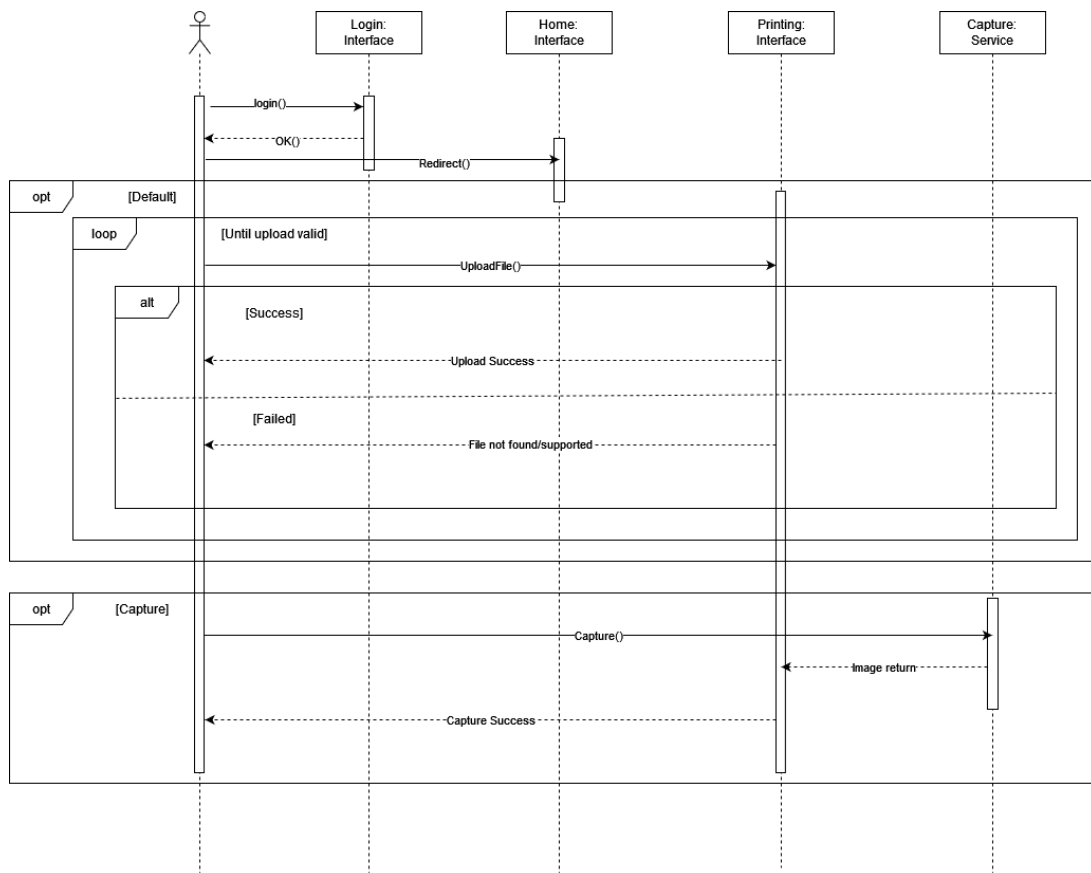
This sequence diagram illustrates the process of configuring and confirming a payment between a user, a payment interface, and a payment service. The interaction begins with the user configuring their order through the "Configure Order()" action directed towards the payment interface. Once the order is configured, the interface proceeds to send a "Confirm Payment()" message to the payment service. The service then initiates a "Send Purchase()" action, which returns a confirmation message back to the interface. At this point, an alternative flow is introduced to handle the response: if the outcome is positive, represented by "[Positive Pages]," the interface proceeds to update the user's credit by invoking "UpdateCredit()" and subsequently presents positive pages. If the response is negative, indicated by "[Negative Pages]," a "Failure Payment" message is sent back to the user, displaying negative pages. This diagram effectively models the flow of a typical payment process, covering both successful and failed scenarios.

3.2.2 Diagram of “Print”

**Description:**

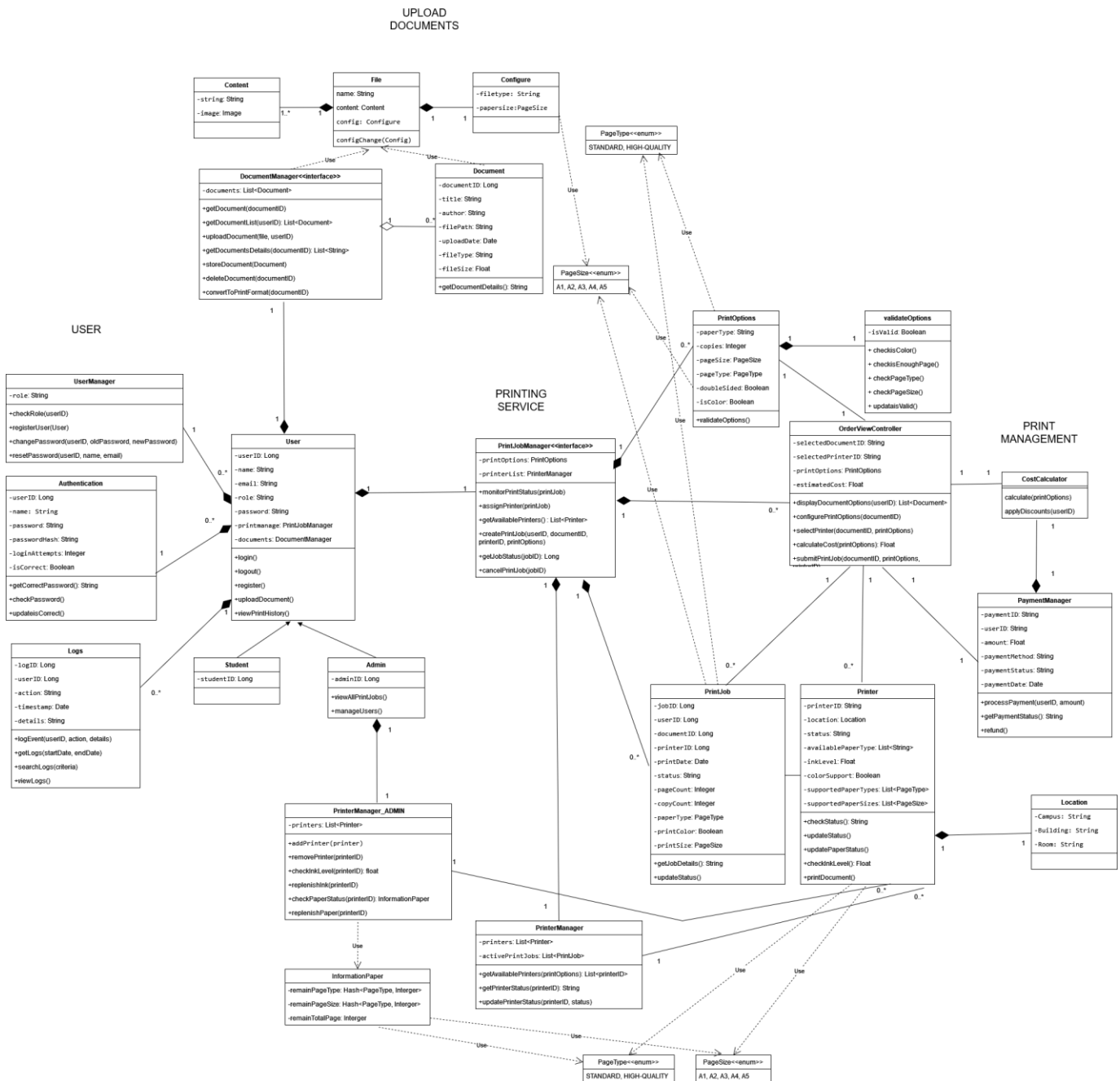
This sequence diagram depicts the interactions between a user, the Printing Interface, and the Printing Service during a file upload and configuration process for printing. The user initiates the process by uploading a file and setting configuration options through the Printing Interface. An alternative flow is presented based on the user's response to the configuration. If the user accepts the configuration, the Printing Interface sends the file to the Printing Service, which processes it and returns a "Print Success" message. If the user rejects the configuration, the configuration settings are saved without sending the file for printing. This diagram highlights the decision-making process and the resulting actions within the printing workflow.

3.2.3 Diagram of “Upload documents”

**Description:**

This sequence diagram illustrates the interaction between a user, various interfaces (Login, Home, Printing), and a Capture Service within an application. The process begins with the user logging in via the Login Interface, which confirms the login and redirects them to the Home Interface. In the "Default" optional flow, the system enters a loop where the user repeatedly attempts to upload a file through the Home Interface until a valid upload is detected. If the upload succeeds, a success message is returned; otherwise, a failure message indicates an unsupported file type or a missing file. Additionally, an optional "Capture" flow is shown where the user can initiate a capture operation through the Printing Interface, which interacts with the Capture Service to return an image, indicating a successful capture process. This diagram provides an overview of the upload and capture workflows, depicting possible outcomes and interactions across components.

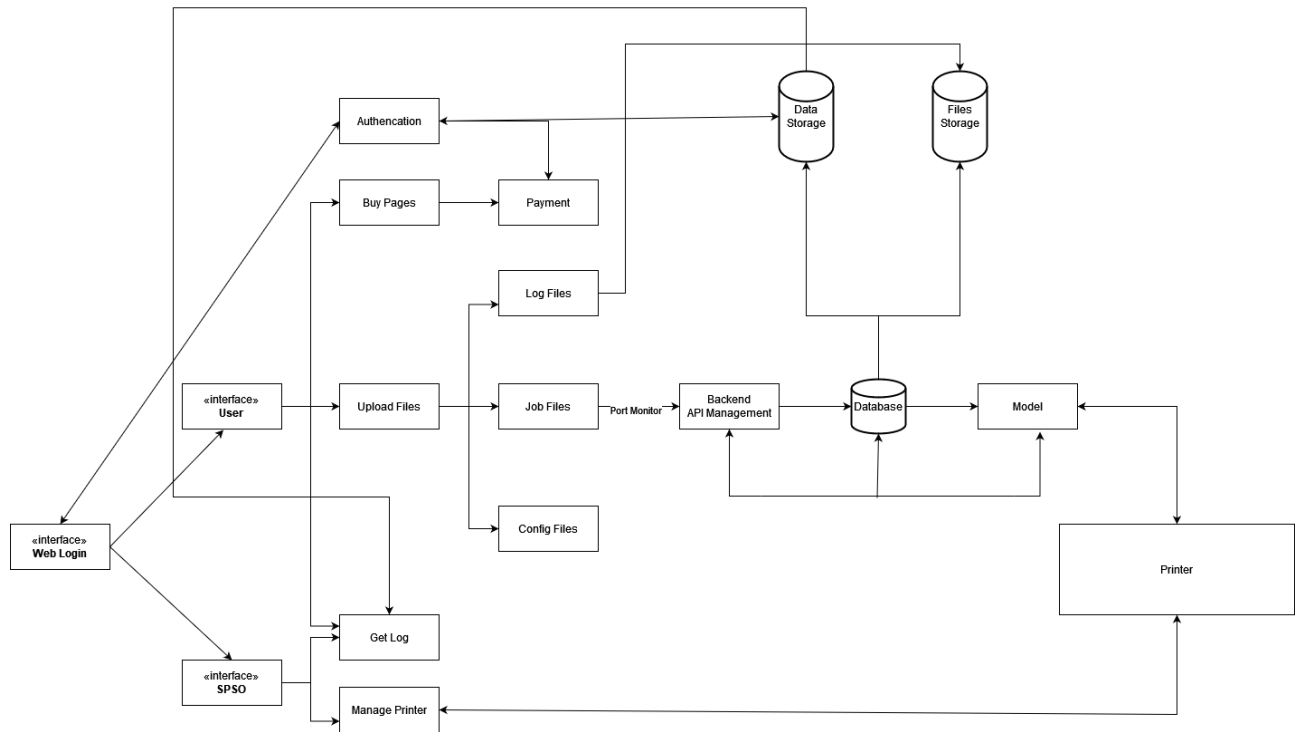
3.3 Class Diagram



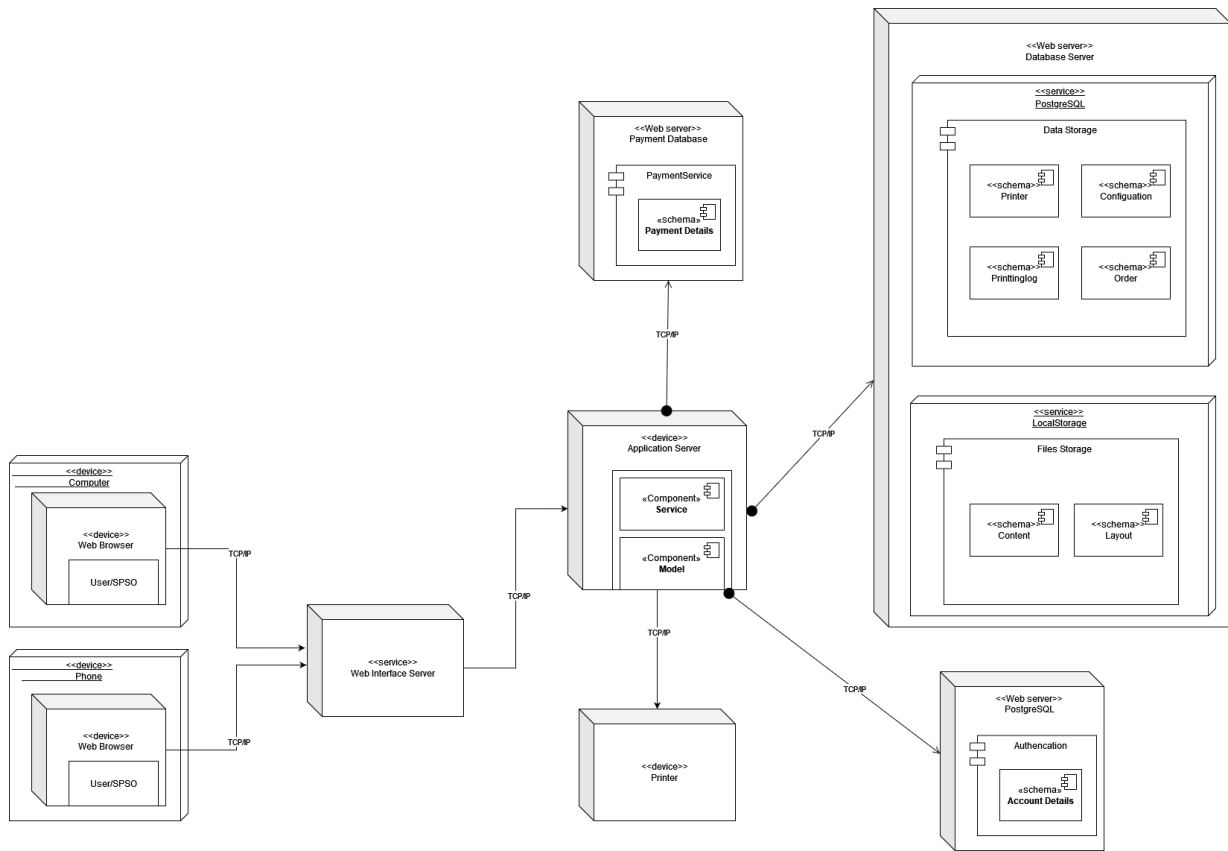
4. Architecture Design

4.1 Architectural Diagram

4.1.1 Architecture



4.1.2 Deployment



4.1.3 Presentation strategy:

- The SSPS software will be designed as a Single Page Application (SPA) to ensure that accessing the application only requires a single page load. This approach improves the speed of user interactions, enhancing the user experience. This will be implemented using React/NextJS, which are popular and widely recognized frameworks within the developer community, making them easy and effective to use.
- To display a page within the application, each page will have a Layout layer with a structure that includes a “Header,” “Main Body,” and “Footer.” This helps developers reuse components like the “Header” and “Footer,” reducing development time and promoting consistency.
- Each page in the application will correspond to a “page” in the source code, and each page will contain components (for example, the “User Page” will have components like “navbar,” “info,” “avatar,” etc.). This approach breaks the source code into smaller parts, creating a robust structure with benefits such as easy operation, maintainability, and reusability.

- Finally, a “model” layer will be established to facilitate data collection when communicating with the backend server. When developers need data to render a page, they place server interactions (often called business logic) within the “model” and call on them when needed. This structure makes it easier to manage and maintain complex functions that require frequent updates.
- Overall, the application incorporates many principles of the MVC (Model-View-Control) model, aiming to optimize the development process, increase efficiency, and minimize costs.

4.1.4 Data storage:

The application data will be divided into two types: informational data (data such as user information, print counts, etc.) and files (print images, Word/PDF files uploaded by users).

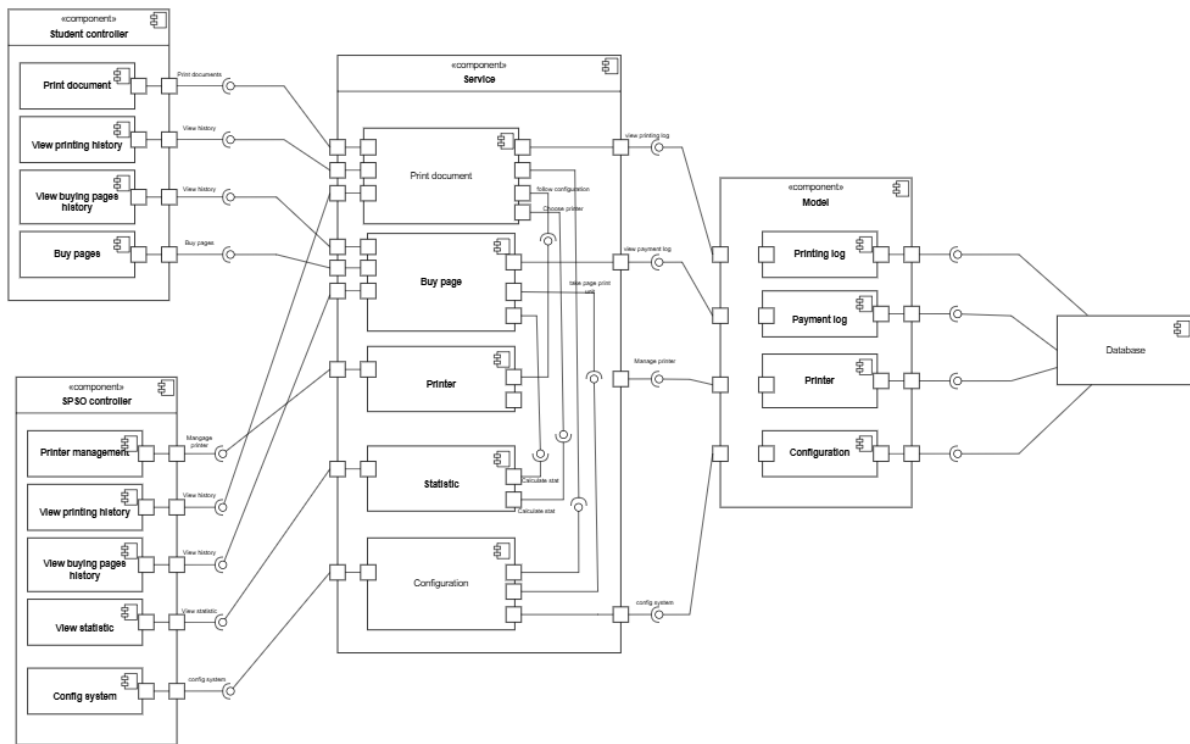
- Data Storage: Database choice: SSPS chooses a relational database, PostgreSQL, to structurally store user data, such as user information, print counts, and more. PostgreSQL has been a popular SQL database in recent years and has good scalability.
 - The PostgreSQL database system will store information across two main databases: the User Database (basic user information) and the Printing Database (storing the addresses of user images or files, related parameters like page count, print type, etc.).
- File Storage: The system will primarily store image files and PDFs in local storage on the server. Most of the files that users request for printing will not need to be stored; users will generally manage these files themselves. Using local storage enhances query speed, convenience, and—importantly—saves unnecessary costs.
 - For cases requiring file storage, SSPS allows users to register for file storage in the system’s 10GB disk memory. SSPS will apply a "storage lending" mechanism, allowing files to exist in the system only for a limited time before being removed to free up space. This time limit will be surveyed and continuously updated during development and usage to balance disk size with student needs.

4.1.5 API management

- To achieve efficiency between the Frontend and Backend systems, the project needs a method to present the existing services of the Backend so that the Frontend has full information on the available services.
- The project has chosen NestJS as the framework for the Backend, and OpenAPI is an excellent choice for this framework. OpenAPI is a library developed by NestJS, utilizing SwaggerAPI as the main tool to manage the available API services in the system.
- Benefits of OpenAPI:
 - Easy to use due to NestJS’s dependency injection mechanism, making it easy to integrate Swagger into the codebase without causing code sprawl (loss of code structure).

- Provides many execution capabilities, such as service return types (200 OK, 404 Not Found, etc.).
- Offers a clear schema, so Frontend developers know the exact content they need to send as well as the structure of the returned data.

4.2 Component Diagram



The system will consist of four main components:

1. **Student Controller Component and SPSO Controller Component:**
 - **SPSO Controller:**
 - Manages printers.
 - Views purchase history.
 - Views print history.
 - Manages system configuration.
 - Views statistics.
 - **Student Controller:**
 - Prints documents.
 - Views print history.
 - Views page purchase history.
 - Purchases pages.
 - These controllers will interact and receive data from the database through corresponding APIs to display information on the interface via the Service layer.
2. **Service Component:**
 - Contains service components that provide functions to perform system logic calculations and interact with models for data processing.
3. **Model Component:**

- Contains smaller model components, including: printing log, payment log, printer, and configuration.
- The model components will connect to the database for data processing.