VIETNAM NATIONAL UNIVERSITY - HO CHI MINH CITY

HO CHI MINH CITY UNIVERSITY OF TECHNOLOGY

FACULTY OF COMPUTER SCIENCE AND ENGINEERING

SOFTWARE ENGINEERING - CO3001

ASSIGNMENT REPORT

TASK 2

Vũ Minh Quân - 2152926
Nguyễn Huỳnh An - 2154018
Lương Lê Long Vũ – 2153980
Trần Gia Huy – 2152600
Hoàng Đại Nam – 2153594
Nguyễn Trương Trọng Phúc – 2152246
Phan Ngọc Khải – 2154019

October 2023

Contents

1 System modeling

1.1 Activity diagram

### 1.1.1 Authentication User

It all kicks off when students wants to access the system. They start by providing their login credentials, typically a student ID and password. The system steps in to validate these details. If the information checks out, the system waves the user through, granting access to the desired features. But if there's a mismatch, the system puts up a sign, denying access.
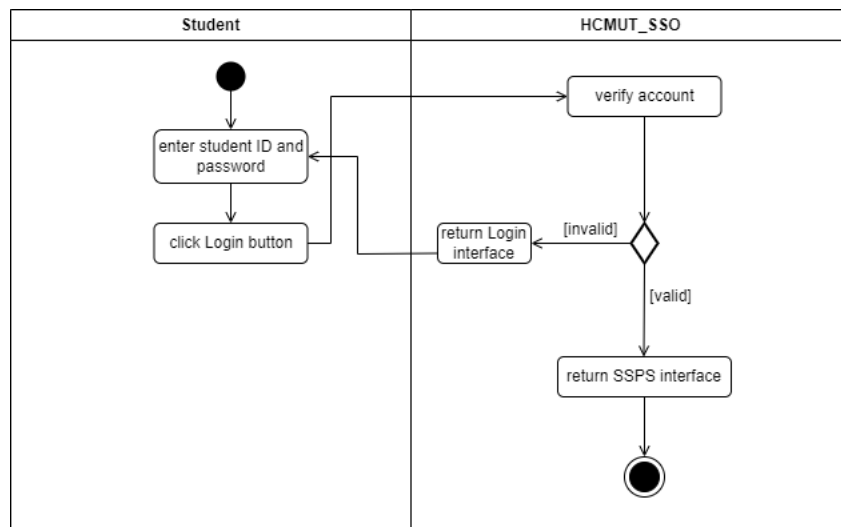
1.1.2 Buy paper

It starts when the student initiates a purchase, specifying their paper preferences, such as type and quantity. The system takes over, handling the user's request and often involving an online payment via a platform like BKPay. After a successful transaction, the system confirms the purchase, and the paper is promptly delivered to the user's satisfaction.
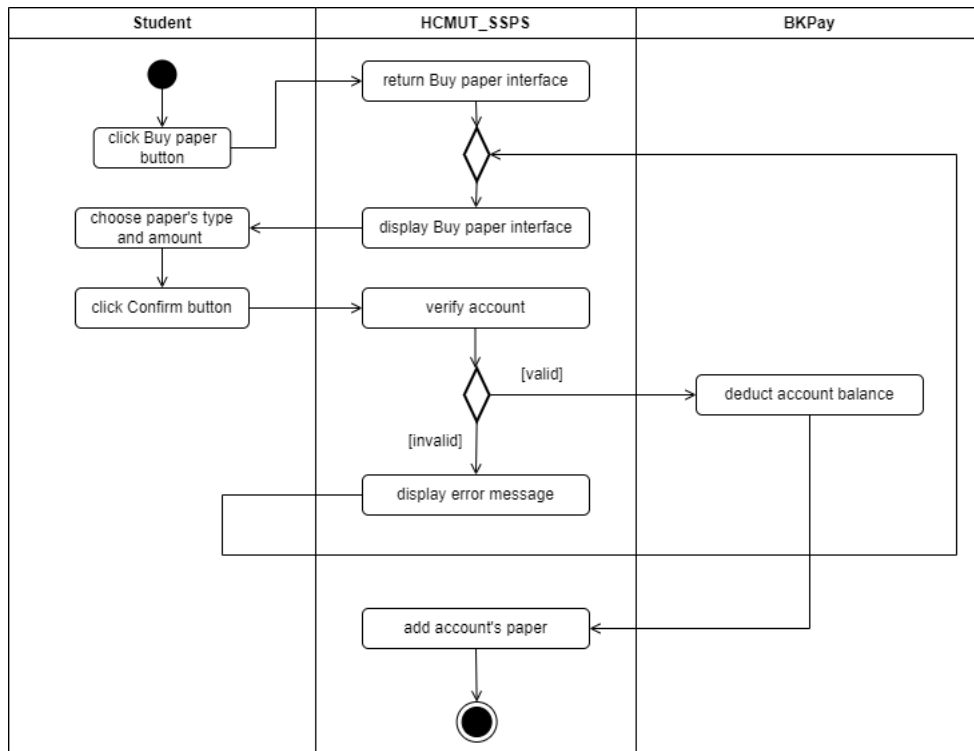
1.1.3 Access printing history

It begins with the user initiating the request, inputting specific parameters. The system then processes the query, retrieving the printing history data from its records. Subsequently, the data is displayed to the user, granting them access to their printing history.
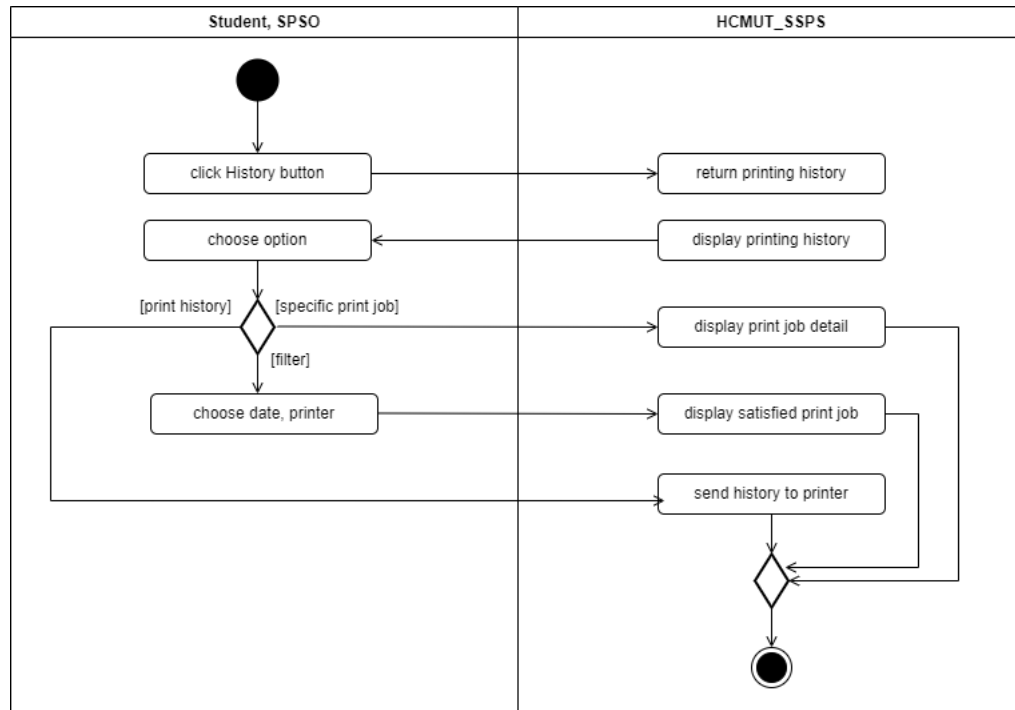
1.1.4 Print files

It kicks off when the student initiates printing, selecting a printer and specifying print properties such as paper size and number of copies. The system takes over, processing the student's request, and then sends the print job to the chosen printer. The printer, in turn, completes the printing task.
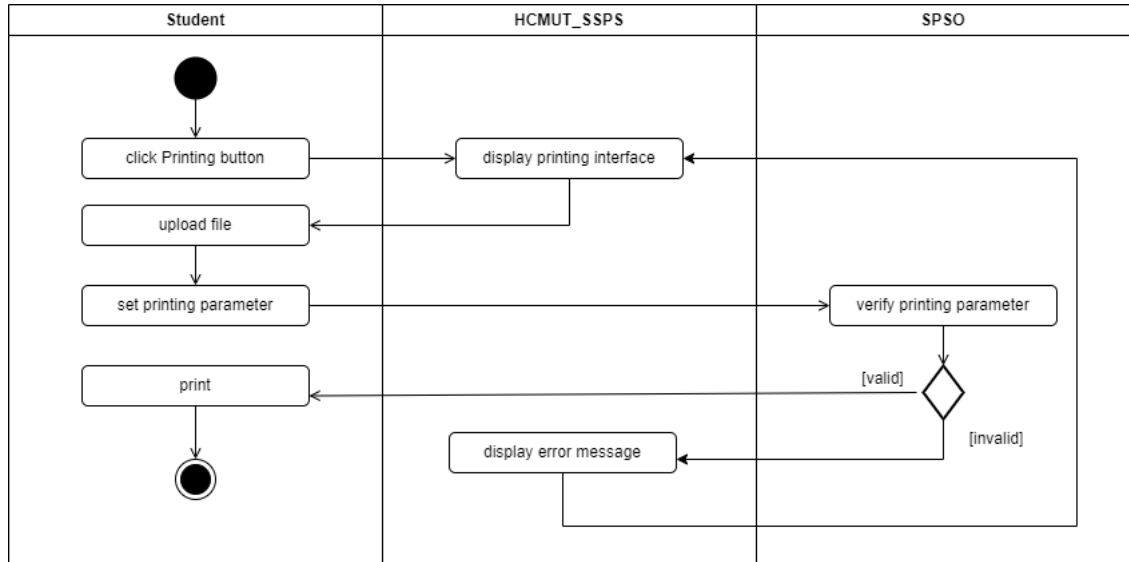
1.2 Sequence diagram

1.2.1 Authentication User

This sequence diagram details the authentication user module of the system. Whenever students enter an ID and password to login the system. The database will verify the account and send the result to HCMUT SSO and system to return the suitable interface.
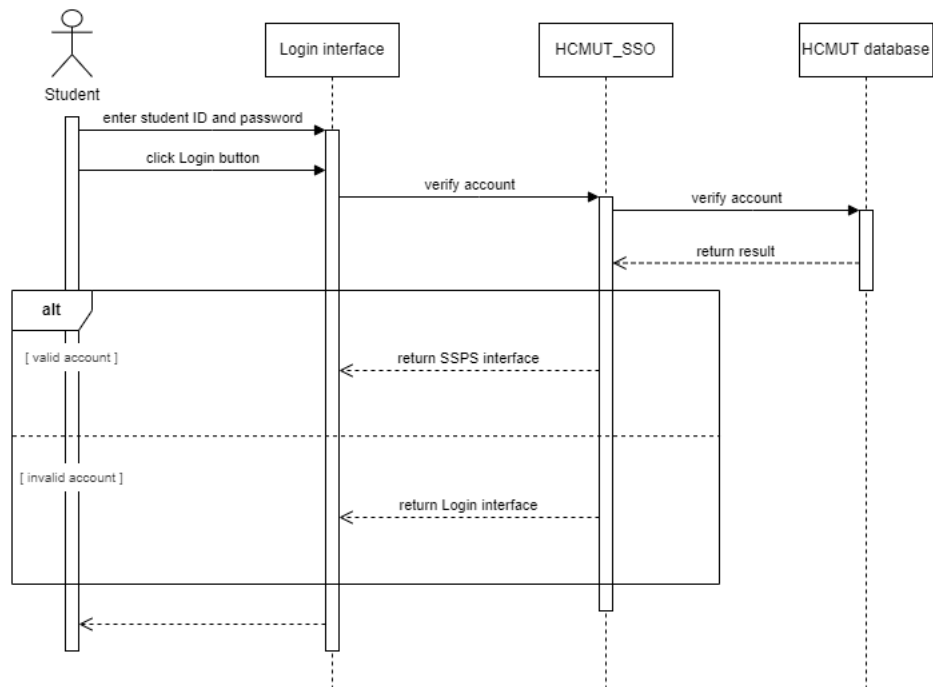
1.2.2 Buy paper

The sequence diagram describes how students buy paper. First, students need to click the Buy paper button and SSPS will display the Buy paper interface. Then, students can choose the printer, the type and amount of paper. Next, the BKPay system will verify the account of the student. If the account is valid, BKPay will deduct the student's account balance and add paper to their account. Otherwise, the system will send an error message to students.
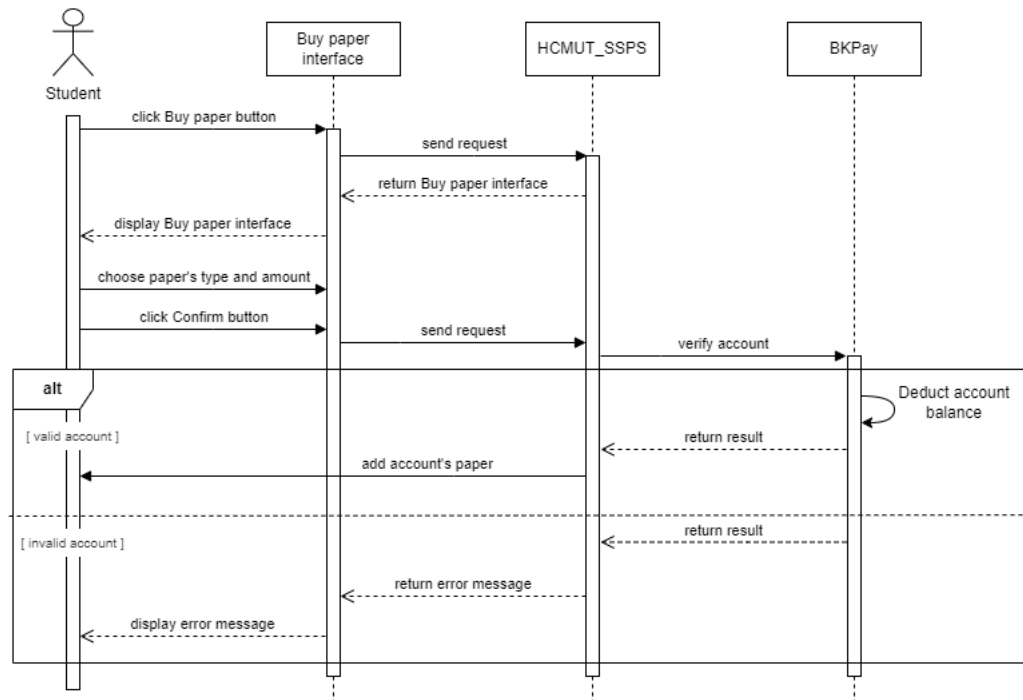
1.2.3 Access printing history

This diagram shows how students access printing history. Whenever students click History button, the system will display History interface. In this interface, students have three options:

1. Accessing detail of print job by clicking specific print job

2. Choosing date or printer to filter print job

3. Printing history
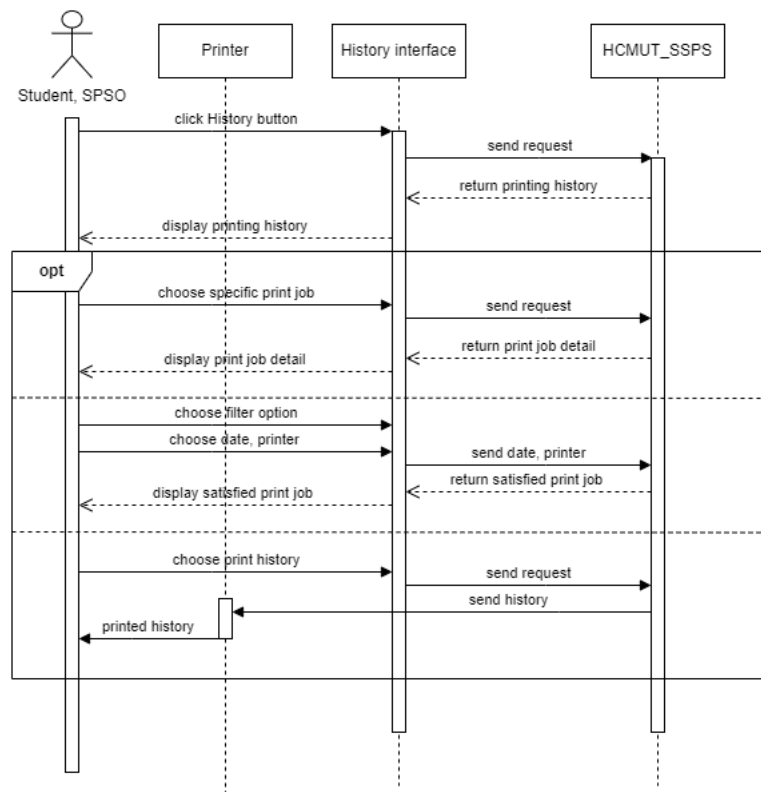
1.2.4 Print files

The Print sequence diagram describes the way to print a file. After clicking Printing button, student can upload file and set parameter to print that file. This file will be verified by the Student Printing Service Officer (SPSO). If this file is valid, the system will send it to printer and student can receive their document. Otherwise, the system will send an error message to students.
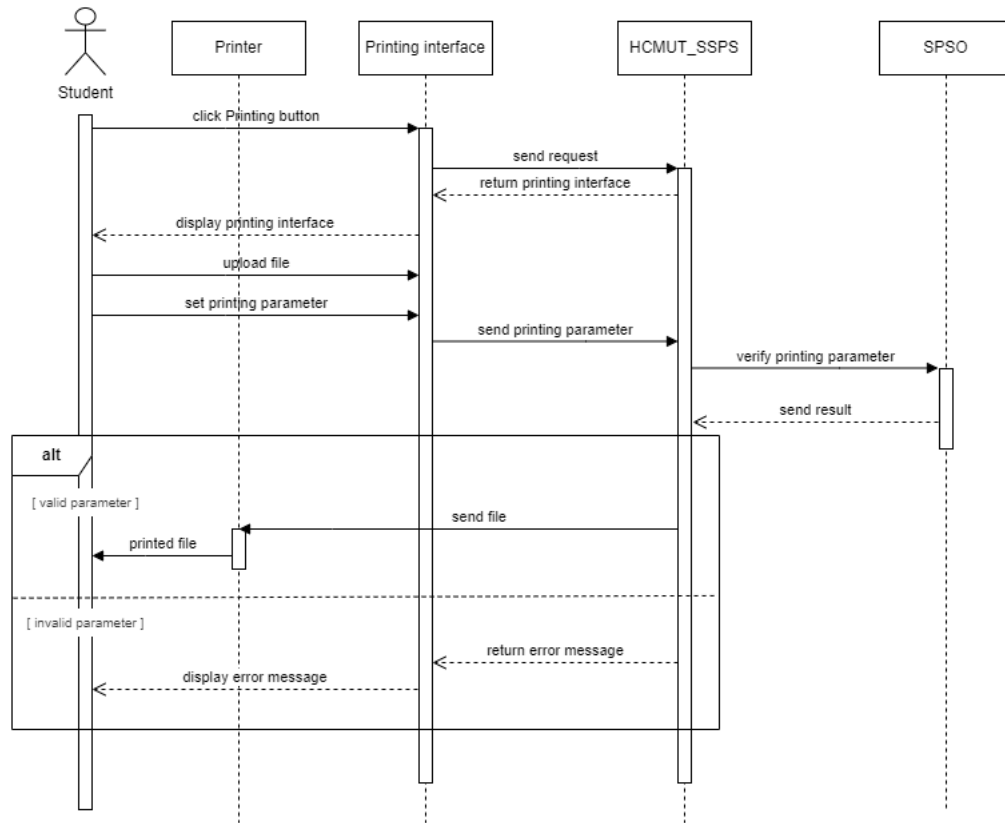
1.3 Class diagram

1.3.1 Authentication User

This class diagram represents a user authentication and information system for HCMUT. The UserAccount class contains details like studentID and password, and it has a direct association with the UserInfo class that stores personal details such as name, email, and address. Users can perform actions like login, logout, and account creation. The HCMUT SSO Service class provides authentication services, maintaining an authToken for validated sessions and facilitating user info requests. User accounts utilize this service for authentication and data retrieval.
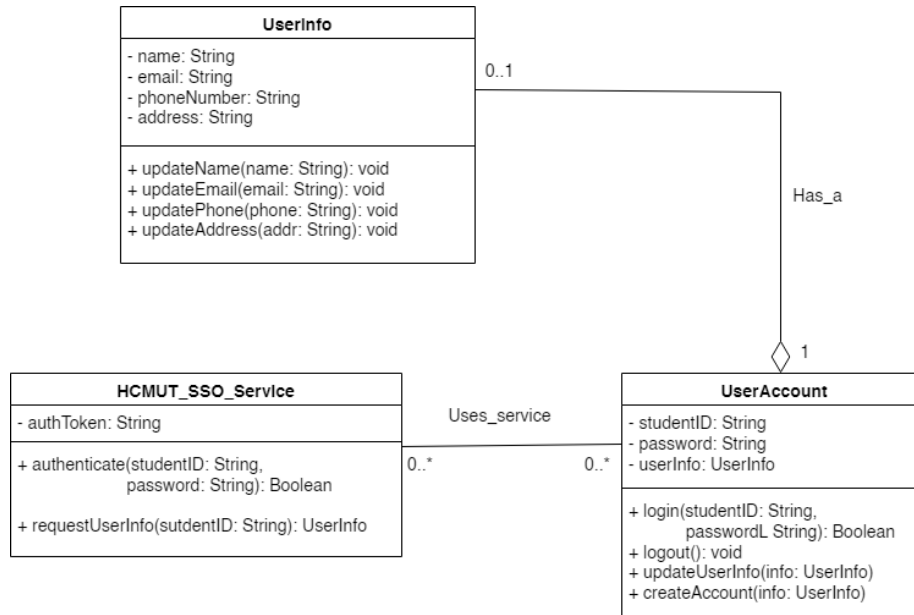
1.3.2 Buy paper

This class diagram represents the "Buy Paper" functionality. The Student class interacts with its Preferences regarding paper type and quantity. They select from the PaperType, which holds details about the paper, its size, price, and inventory. For purchasing, the Student makes a Payment, which uses a specific PaymentMethod (e.g., Credit Card). Once the payment is processed, a TransactionConfirmation is generated, detailing the purchase and confirming the transaction.

Preferences

- preferenceName: String
- selectedPaperType: PaperType
- selectedQuantity: Integer
- lastModified: Date

+ savePreferences(student: Student): void
+ retrievePreferences(studentID: String): Preferences
+ updatePreferences(newPaperType: PaperType, newQuantity: Integer): void
+ deletePreferences(): void

0..1 have   1 selectedIn

1   1

Student

- studentID: String
- name: String
- payment: PaymentMethod
- savedPreferences: Preference

+ selectPaperType(paperType: PaperType): void
+ specifyQUantity(quantity: Integer): void
+ providePaymentDetails(payment: PaymentMethod): Boolean
+ savePreferences(paperType: PaperType, quantity: Integer): void
+ viewTransactionHIstory(): TransactionConfirmation

PaperType

- typeName: String
- size: String (e.g., "A4","A3")
- pricePerUnit: Double
- availableInventory: Integer

+ getAvailableTypes():PaperType
+ checkInventory(quantity: Integer): Boolean
+ updateInventory(amount: Integer): void
+ getPrice(quantity: Integer): Double

select   1   0..*

1   make

TransactionConfirmation

- transactionID: String
- date: Date
- amount: Double
- paymentMethod: PaymentMethod
- status: String

+ displayConfirmation(): void
+ emailConfirmation(email: String): void

Payment

- transactionID: String
- amount: Double
- paymentMethod: PaymentMethod
- status:String (e.g., "success","Failed"

+ processPayment(student: Student, amount Double): Boolean
+ generateConfirmation(): TransactionConfirmation
+ refund():Boolean

use   1   1

PaymentMethod

- methodName: String (e.g., "Credit Card","BKPay"
- accountNumber: String
- expirationDate: Date
- CVB: Integer (only if its a credit card)

+ validatePaymentMethod(): Boolean
+ deductAmount(amount: Double): Boolean
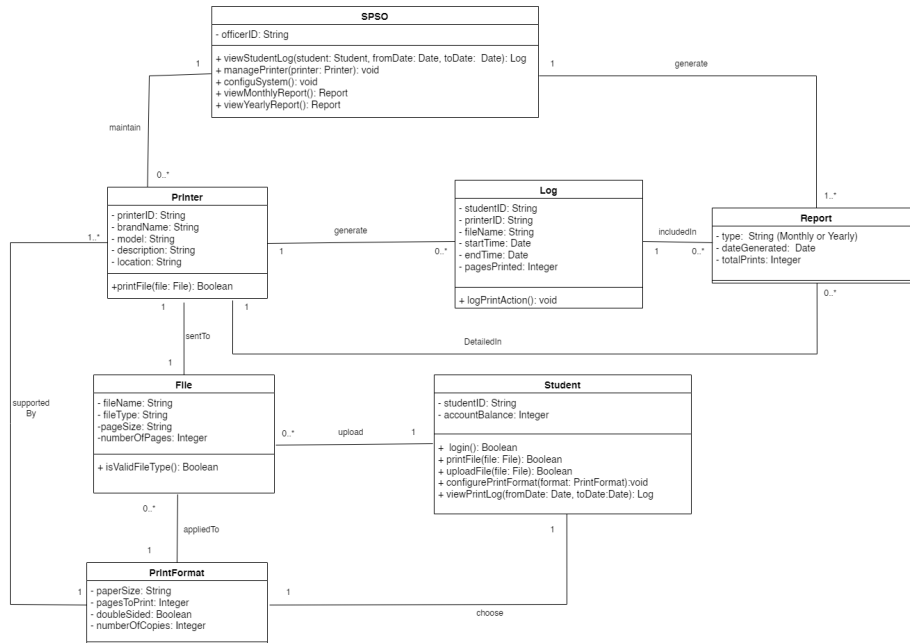
generate   1   1

1.3.3   Access printing history

The class diagram showcases the "Access Printing History" process. A Student can use files, represented by the File class, which contains details like file size and type. The PrintingHistory class captures each printing event, recording details like print date and time. Students can view their print history and also filter it by date, printer, or document. Additionally, the SPSO class represents officers who assist in managing the printing service, while the Printer class lists available printers and their locations.

File

- fileID: Integer
- fileName: String
- fileType: String
- fileSize: Float

+ retrieveFileDetails(fileName: String): File

0..*

contain

1

Student

- studentID: Integer
- name: String
- email: String
- password: String
- loggedIn: Boolean

+ login(email: String, password: String): Boolean
+ viewHistory(): List<PrintingHistory>
+ exportHistory(): File

printingHistory

- historyID: Integer
- printDate: Date
- pageNumbers: Integer
- printingTime: Time

+ filterByDate(date: Date): List<PrintingHistory>
+ filterByPrinter(printer: Printer): List<PrintingHistory>
+ filterByDocument(file: File):List<PrintingHistory>

Use

1   0..*

1

0..*

assisted_by

printed_on

1..*

0..*

SPSO

- officerID: Integer
- officerName: String
- officerEmail: String
- officerShift: String

+ managePrintingService(): void

Printer

- printerID: Integer
- printerName:String
- printerLocation: String

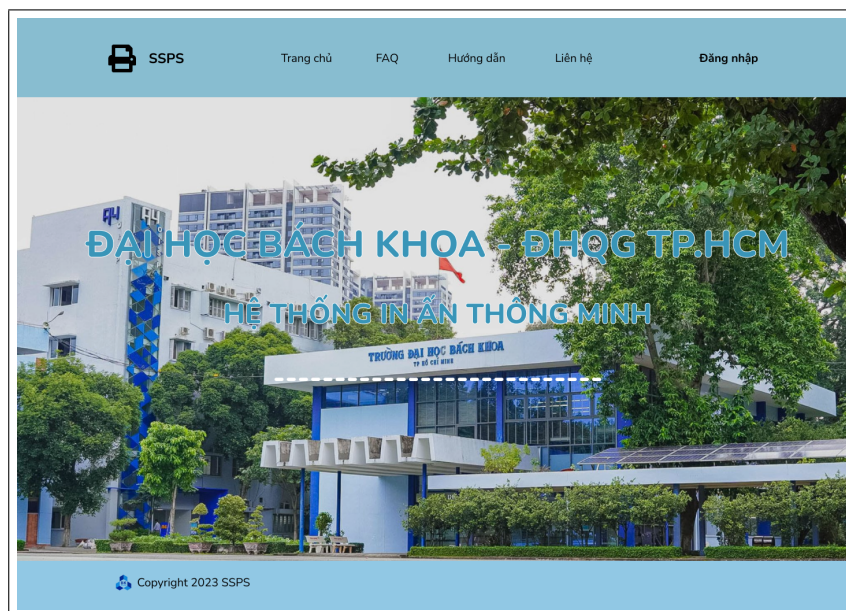+ retrieveAllPrinters():List<Printer>

1.3.4   Print files

The class diagram illustrates the "Print Files" system. Students can upload Files with specific attributes and select a PrintFormat, detailing parameters like paper size and the number of copies. The Printer class represents devices available for printing, each with unique details. SPSO oversees the system, maintaining printers and viewing student logs. Every print action is recorded in the Log class, which can be compiled into monthly or yearly Reports by the system.
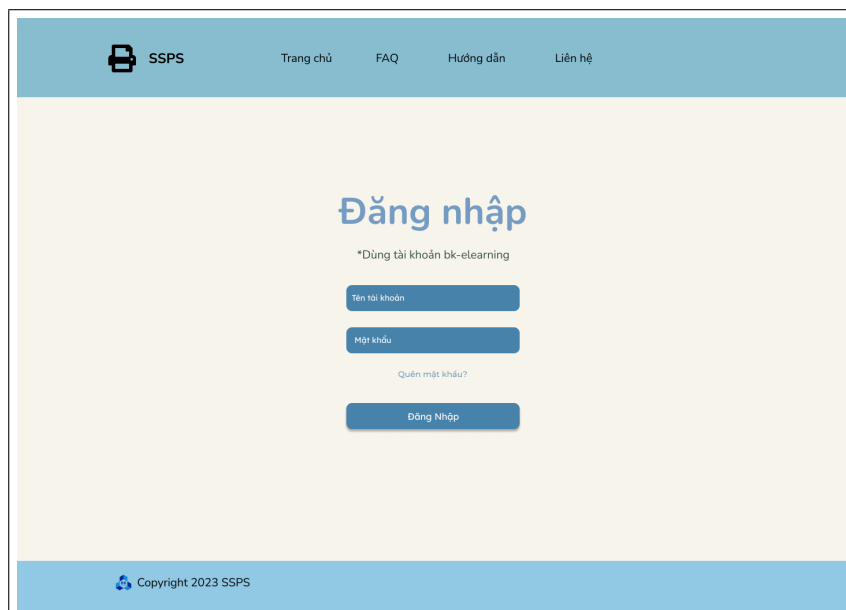
SPSO

- officerID: String

+ viewStudentLog(student: Student, fromDate: Date, toDate: Date): Log
+ managePrinter(printer: Printer): void
+ configuSystem(): void
+ viewMonthlyReport(): Report
+ viewYearlyReport(): Report

Printer

- printerID: String
- brandName: String
- model: String
- description: String
- location: String

+printFile(file: File): Boolean

Log

- studentID: String
- printerID: String
- fileName: String
- startTime: Date
- endTime: Date
- pagesPrinted: Integer

+ logPrintAction(): void

Report

- type: String (Monthly or Yearly)
- dateGenerated: Date
- totalPrints: Integer

File

- fileName: String
- fileType: String
- pageSize: String
- numberOfPages: Integer

+ isValidFileType(): Boolean

Student

- studentID: String
- accountBalance: Integer

+ login(): Boolean
+ printFile(file: File): Boolean
+ uploadFile(file: File): Boolean
+ configurePrintFormat(format: PrintFormat):void
+ viewPrintLog(fromDate: Date, toDate:Date): Log

PrintFormat

- paperSize: String
- pagesToPrint: Integer
- doubleSided: Boolean
- numberOfCopies: Integer

generate

maintain

0..*

1..*

1

generate

1

0..*

includedIn

1

0..*

1..*

0..*

DetailedIn

sentTo

1

1

1

supported
By

0..*

upload

1

1

choose

1

appliedTo

1

0..*

1

1.4 User interface

1.4.1 Homepage

Source: `hcmut.edu.vn`

1.4.2 Login interface

1.4.3 Function Page
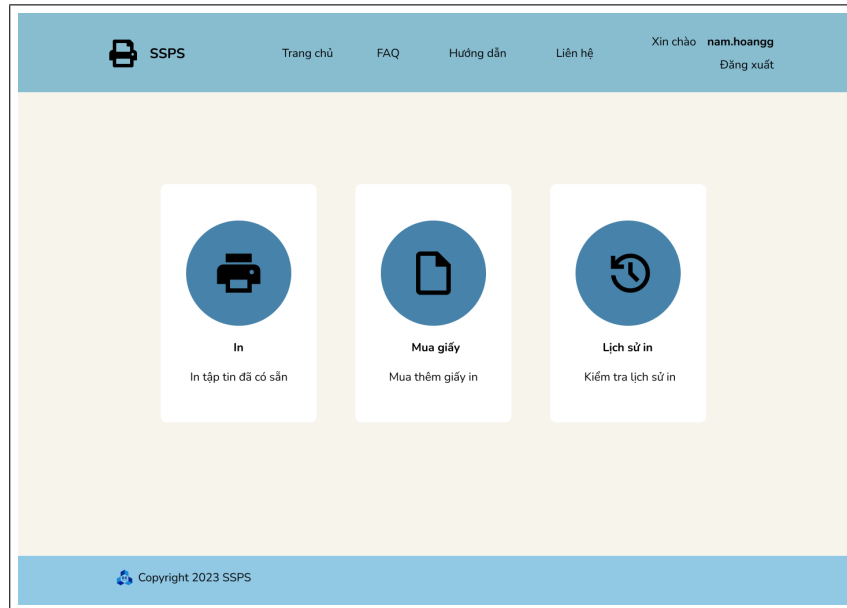
1.4.4 Printing Interface

1.4.5   Buy extra paper

1.4.6   History Page