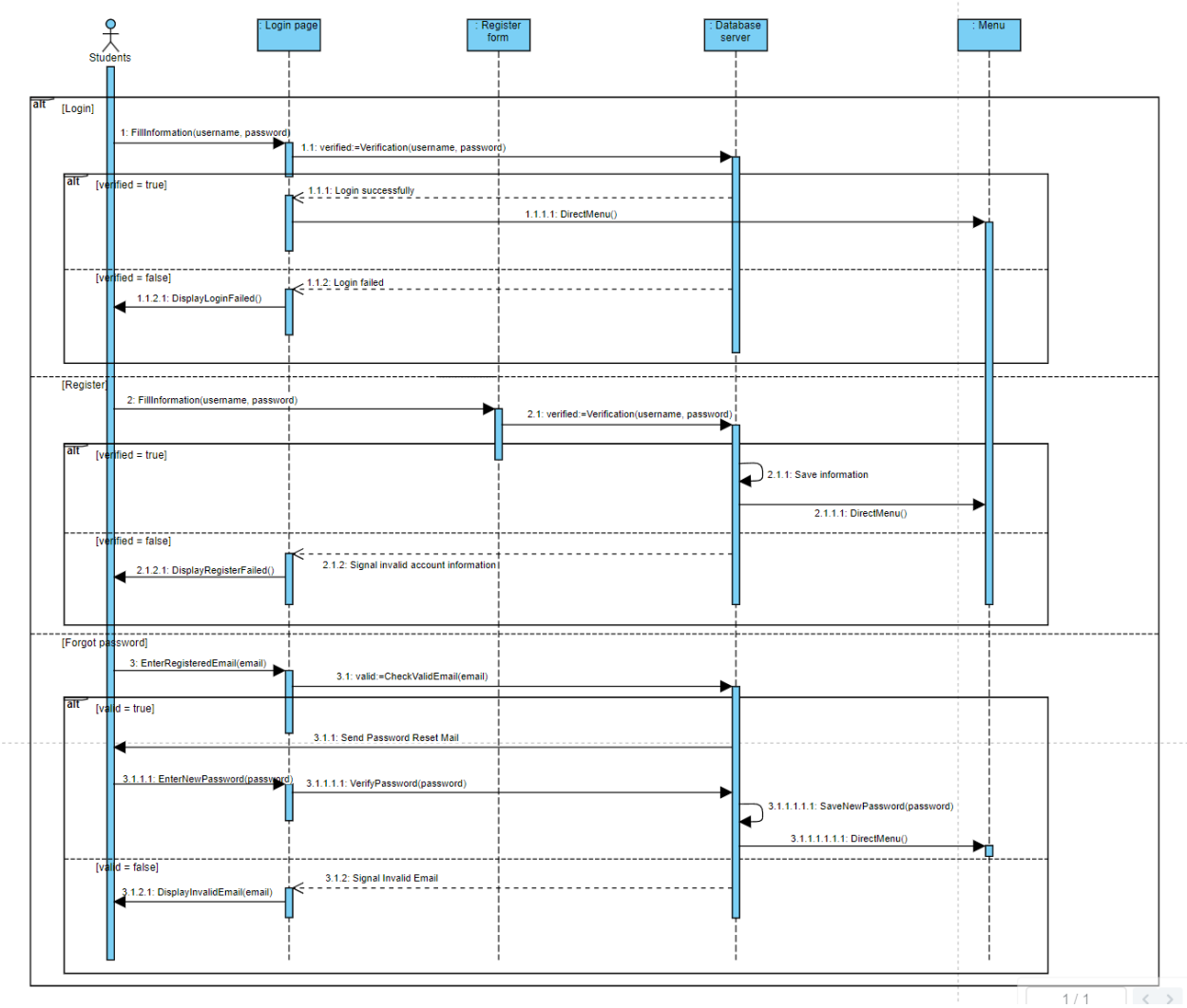
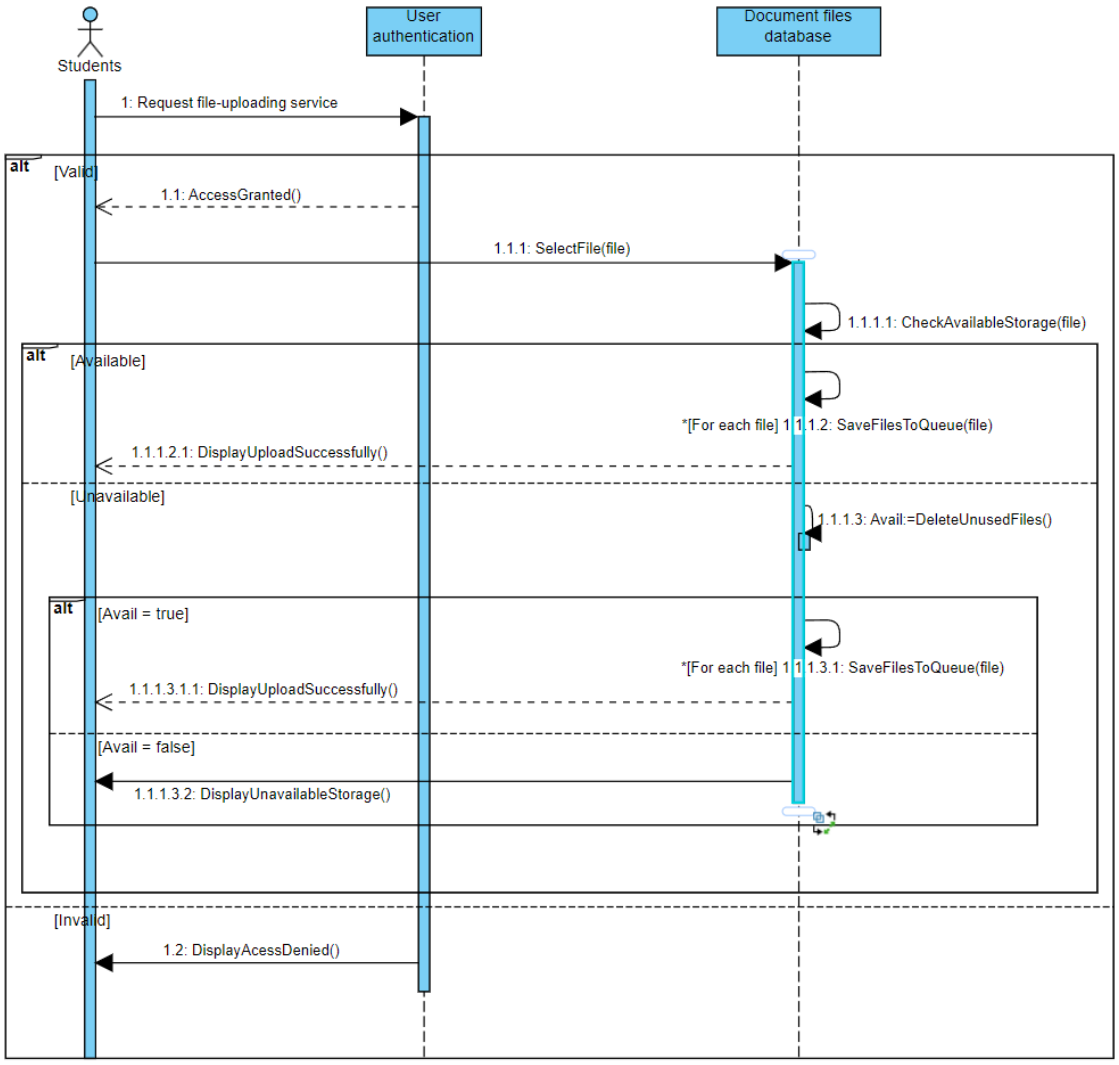


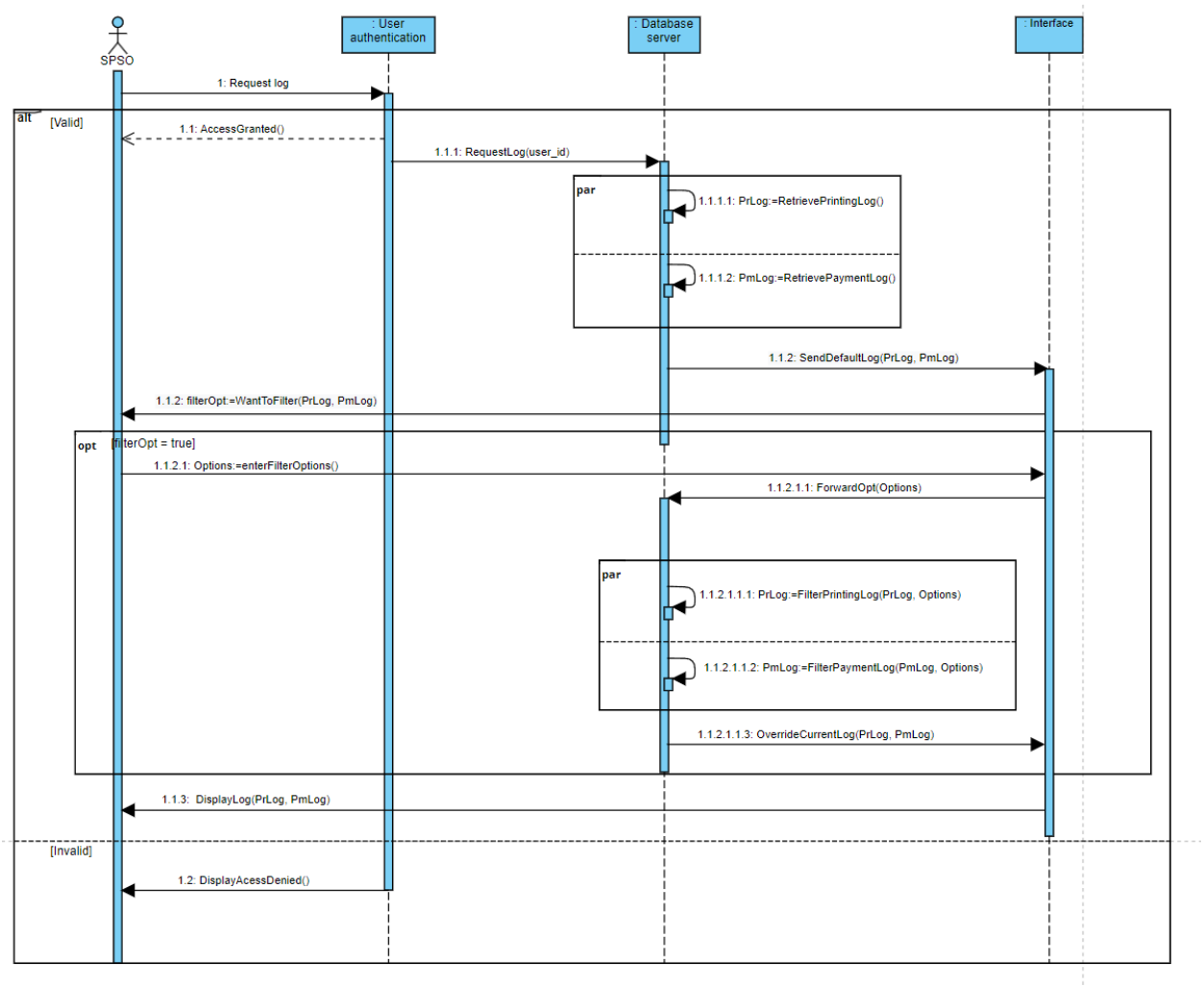
Description:

The above diagram is Login Sequence Diagram. There are 3 main alt fragments: login if the Students actor already has an account, register is the user has not had an account yet and forgot password. In login fragment, the Students actor will enter username and password then the system will verify the information. If the information is valid, the actor will login successfully and will be directed to menu; if it's invalid information, actor will get failed login notification. In register fragment, after filling username and password, if the information is valid, the system will save information in its database, user will login successfully and will be directed to menu, otherwise failed registration notification is displayed. In forgot password, the actor will enter email that has already been registered then system will check if the entered email is valid. If it's a valid email, system will send password-reset email and the actor will enter new password. The system will save the new password and direct successfully login user if the password is valid; if the email is invalid, notification is displayed.



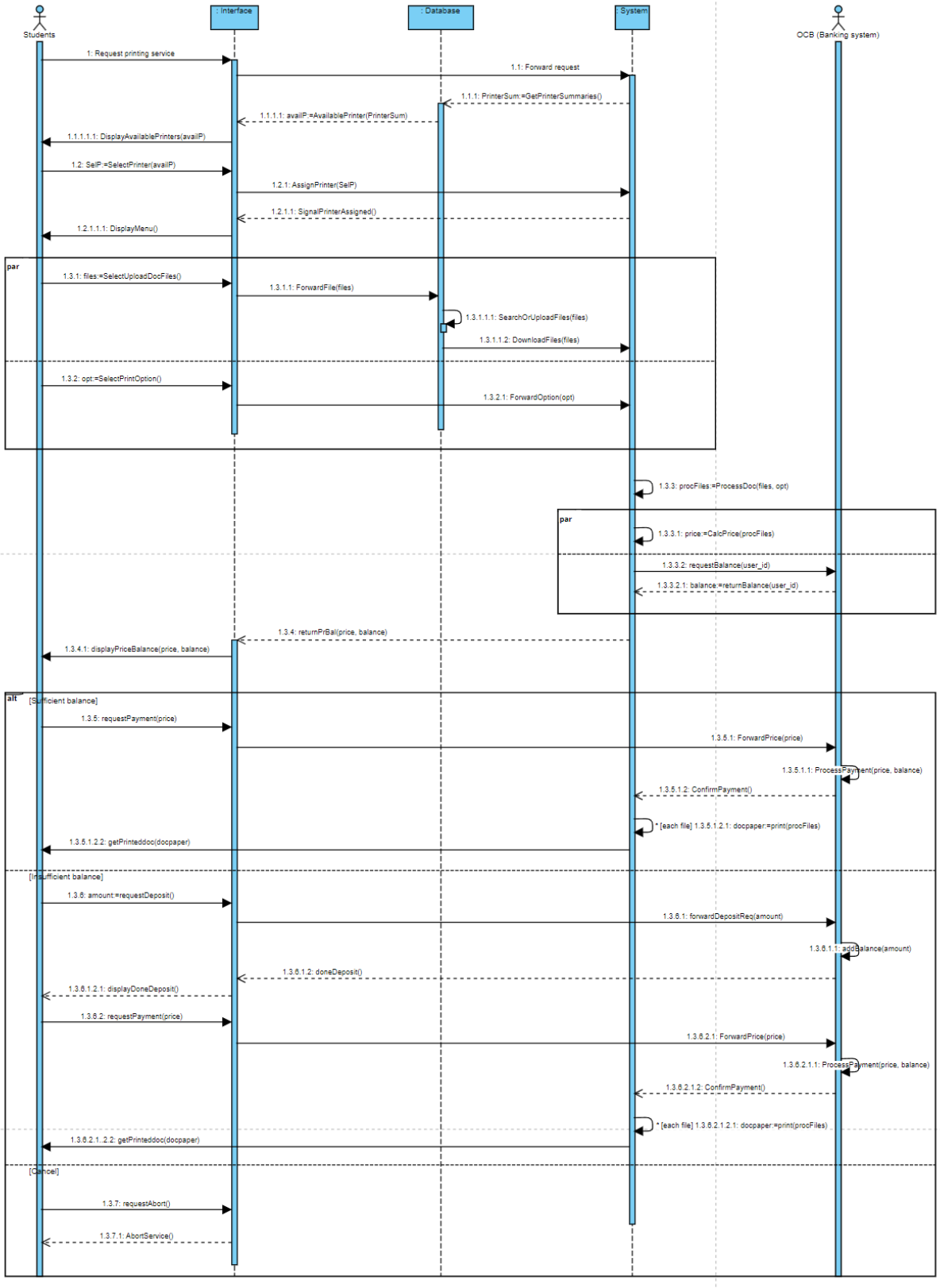
Description:

The above diagram is Upload printing files Sequence Diagram. The Students actor will first request uploading service and system will verify user. If it's an invalid user, access is denied. Otherwise, the actor will be granted access, after that, user can select files to upload and the system database will check if there's available storage for the files. If available, files will be uploaded and saved to queue, success notification is displayed. If unavailable, the database will try to delete unused files. After that, if there's still not enough storage, unavailable storage notification is displayed, otherwise, files will be uploaded and saved to queue, success notification is displayed.



Description:

The above diagram is View log Sequence Diagram. The SPSO actor will request retrieving log and system will verify user. If it's an invalid user, access is denied. Otherwise, the actor will be granted access, after that, system will request log from database with id of user. Next, database server will retrieve both printing log and payment log concurrently, and send default logs to interface. After that, SPSO actor will be asked if they want to filter out the logs. If yes, the actor will enter desired filtering options, then database will filter out both printing log and request log concurrently based on given options. Next, the default log is overridden with the filtered log. Finally, log will be displayed to user.


Students
: Interface
: Database
: System
OCB (Banking system)
1: Request printing service
1.1: Forward request
1.1.1: PrinterSum:=GetPrinterSummaries()
1.1.1.1: availP:=AvailablePrinter(PrinterSum)
1.1.1.1.1: DisplayAvailablePrinters(availP)
1.2: SelP:=SelectPrinter(availP)
1.2.1: AssignPrinter(SelP)
1.2.1.1: SignalPrinterAssigned()
1.2.1.1.1: DisplayMenu()
par
1.3.1: files:=SelectUploadDocFiles()
1.3.1.1: ForwardFile(files)
1.3.1.1.1: SearchOrUploadFiles(files)
1.3.1.1.2: DownloadFiles(files)
1.3.2: opt:=SelectPrintOption()
1.3.2.1: ForwardOption(opt)
1.3.3: procFiles:=ProcessDoc(files, opt)
par
1.3.3.1: price:=CalcPrice(procFiles)
1.3.3.2: requestBalance(user_id)
1.3.3.2.1: balance:=returnBalance(user_id)
1.3.4: returnPrBal(price, balance)
1.3.4.1: displayPriceBalance(price, balance)
alt
[Sufficient balance]
1.3.5: requestPayment(price)
1.3.5.1: ForwardPrice(price)
1.3.5.1.1: ProcessPayment(price, balance)
1.3.5.1.2: ConfirmPayment()
* [each file] 1.3.5.1.2.1: docpaper:=print(procFiles)
1.3.5.1.2.2: getPrinteddoc(docpaper)
[Insufficient balance]
1.3.6: amount:=requestDeposit()
1.3.6.1: forwardDepositReq(amount)
1.3.6.1.1: addBalance(amount)
1.3.6.1.2: doneDeposit()
1.3.6.1.2.1: displayDoneDeposit()
1.3.6.2: requestPayment(price)
1.3.6.2.1: ForwardPrice(price)
1.3.6.2.1.1: ProcessPayment(price, balance)
1.3.6.2.1.2: ConfirmPayment()
* [each file] 1.3.6.2.1.2.1: docpaper:=print(procFiles)
1.3.6.2.1.2.2: getPrinteddoc(docpaper)
[Cancel]
1.3.7: requestAbort()
1.3.7.1: AbortService()

Description:

The above diagram is Print documents Sequence Diagram. The Students actor will request printing service, then after receiving request, system will send to user a list of currently available printers. The actor will select printers so that the system can assign jobs to them and remove them from available queue. Then Students actor can upload or select already uploaded document files and choose printing options. Selected files will be either searched in already uploaded queue or uploaded if the files are new. Then files will be downloaded to printing system and printing options will be forwarded as well. After printing system has received files and printing options, system will process and format documents based on given options. Next, concurrently, price is calculated and current user's balance is requested from OCB actor to display to user. If Students actor chooses to cancel printing service (including if the balance is insufficient and user does not want to deposit money), service is aborted. If balance is insufficient, user will deposit an amount of money to account's balance, then wait for OCB to add money to balance and send confirmation. After that, for both sufficient and insufficient balance cases, user will request payment, wait for OCB to process payment and send confirmation to the system. After receiving payment confirmation, the system will start printing documents. Finally, users will get their printed documents at the locations of previously chosen printers.