1. USER INTERFACE

1.1 Standardizing the screen configuration

Display

Number of colors supported: 16,777,216 colors

Resolution: 1366 x 768 pixels

Screen

Location of standard buttons: At the bottom (vertically) and in the middle (horizontally) of the frame

Location of the messages: Starting from the top vertically and in the middle horizontally of the frame down to the bottom

Display of the screen title: The title is located at the top of the frame in the middle

Consistency in expression of alpha numeric numbers: comma for separator of thousand while strings only consist of characters, digits, commas, dots, spaces, underscores, and hyphen symbol

Control

Size of the text: medium size (mostly 24px). Font: Segoe UI. Color: #000000

Input check process: Should check if it is empty or not. Next, check if the input is in the correct format or not

Sequence of moving the focus: There will be no stack frames. Each screen will be separated. However, the manual is considered a popup message, as the main screen cannot be operated while the manual screen is shown. After submitting the delivery form, in case the user chooses to place rush order, the app will move to rush order screen. After the rush order screen, the app will move to the invoice screen

Sequence of the system screens:

1. Rush order screen

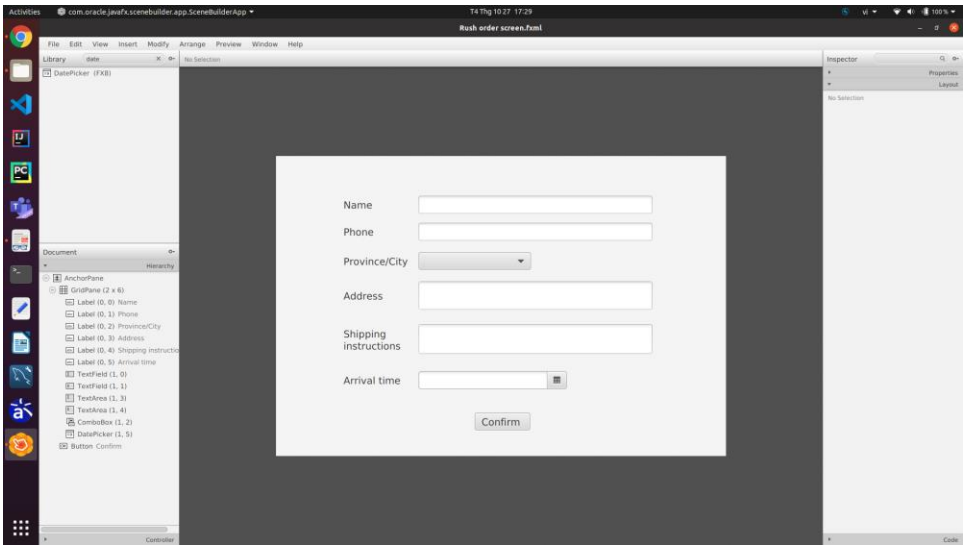
Direct input from the keyboard

There will be no shortcuts. There are back buttons to move back to the previous screen. Also, there is the close button “X” located at the title bar to the right to close the screen.

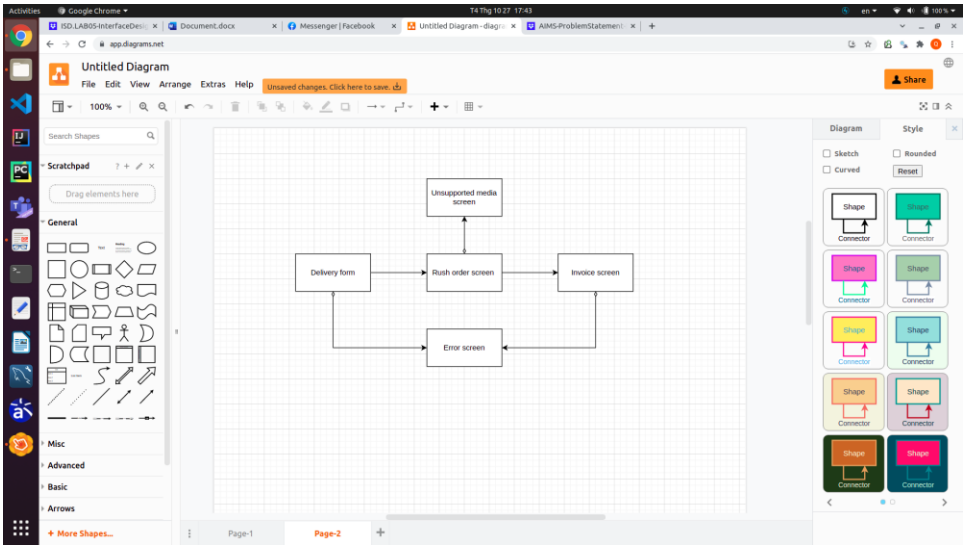
Error

A message will be given to notify the users what the problem is.

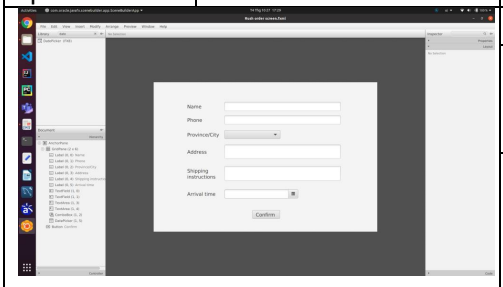
1.2 Creating screen images



1.3 Creating a screen transition diagram

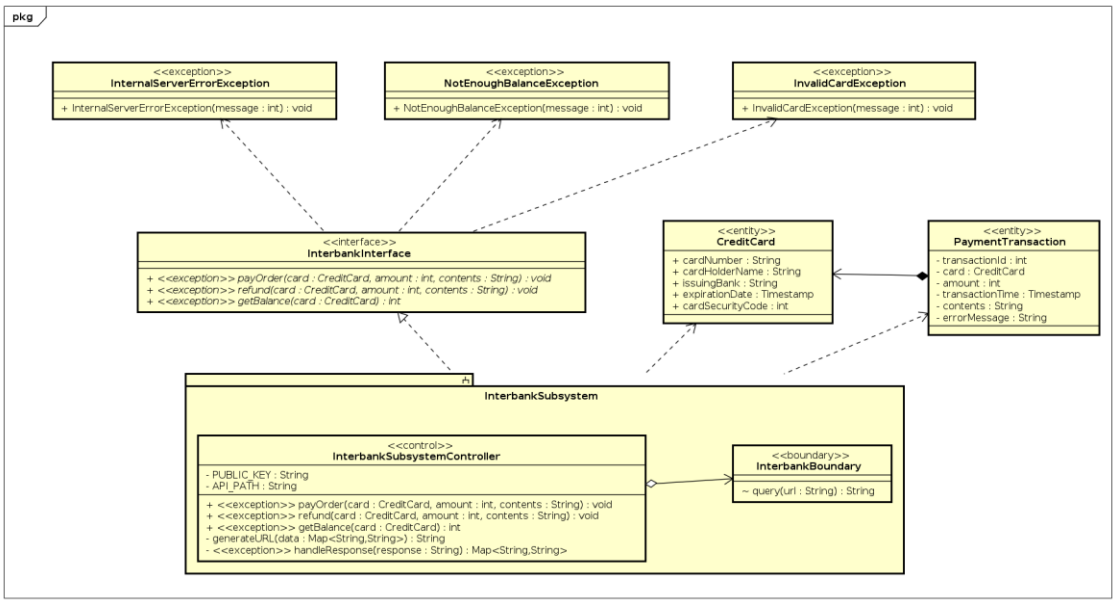


1.4 Creating screen specification

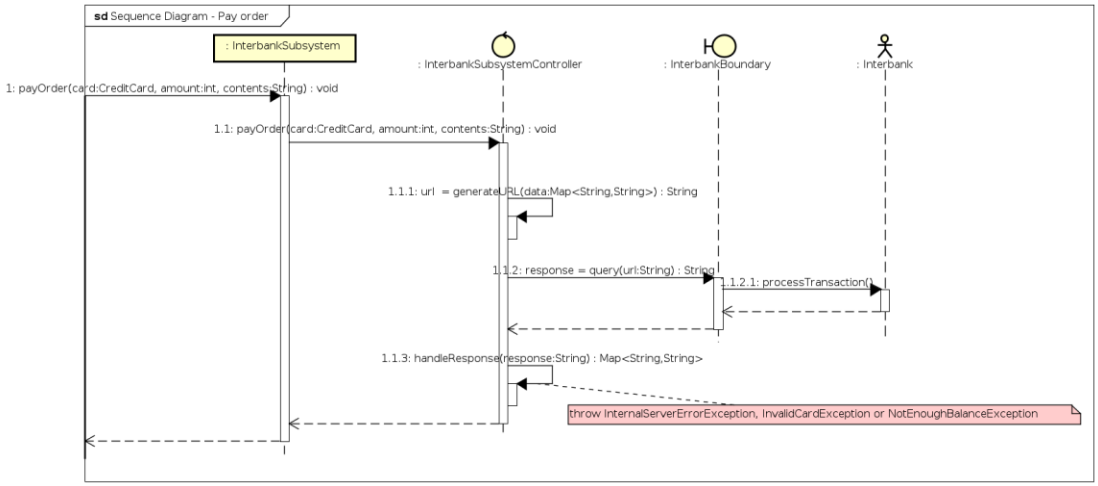
AIMS Software		Date of creation	Approved by	Reviewed by	Person in charge
Screen specification	Place rush order form	27/10/2021			Ta Minh Quan
		Control	Operation	Funciton	
		Area for showing rush order	Initial	Display the form with blanks for user to enter the info	
		Confirm	Click	Confirm that the delivery info for the rush order is correct	

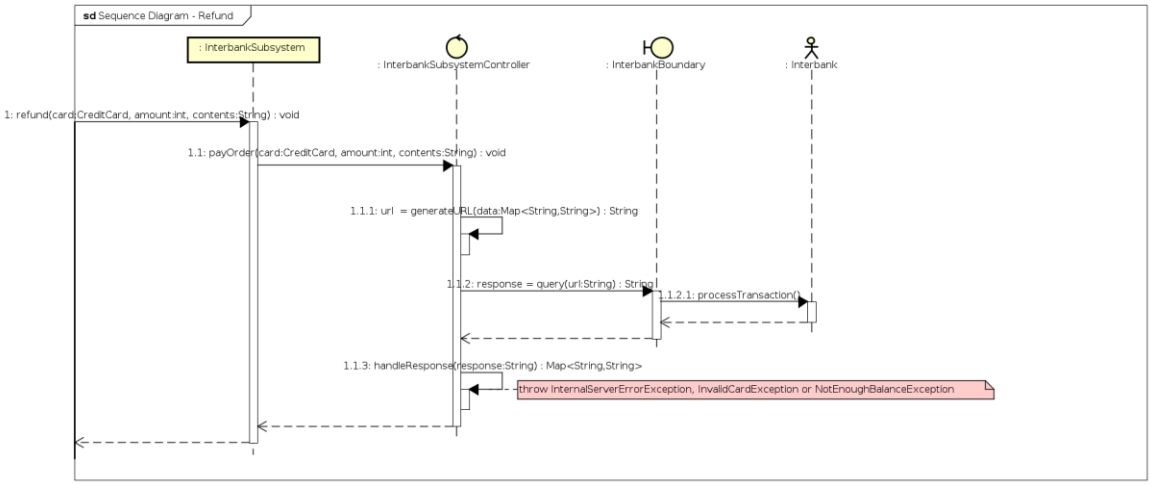
2. INTERFACE DESIGN

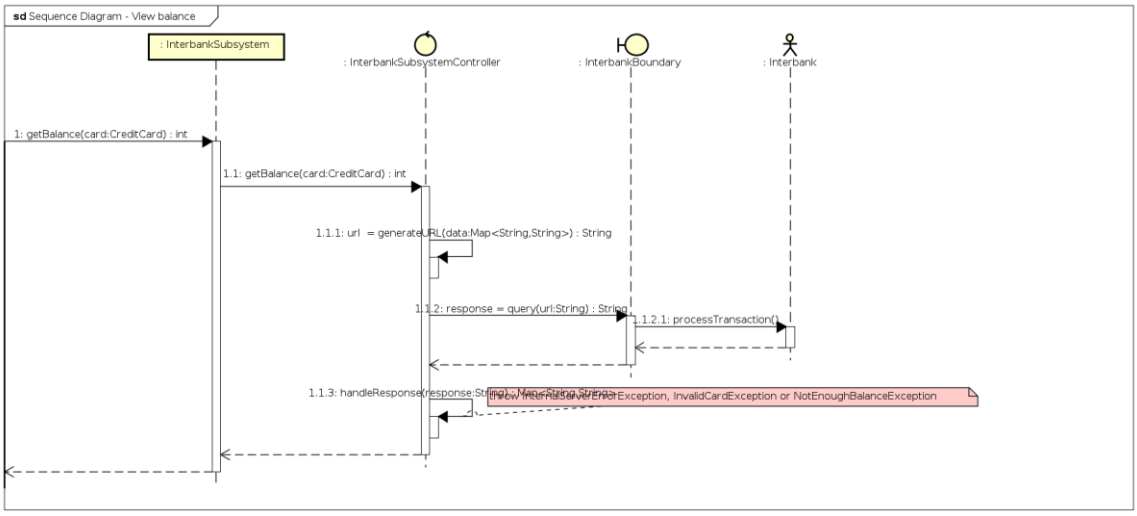
2.1 Class diagram



2.2 Sequence diagram




sd Sequence Diagram - Refund
: InterbankSubsystem
: InterbankSubsystemController
: InterbankBoundary
: Interbank
1: refund(card:CreditCard, amount:int, contents:String) : void
1.1: payOrder(card:CreditCard, amount:int, contents:String) : void
1.1.1: url = generateURL(data:Map<String,String>) : String
1.1.2: response = query(url:String) : String
1.1.2.1: processTransaction()
1.1.3: handleResponse(response:String) : Map<String,String>
throw InternalServerErrorException, InvalidCardException or NotEnoughBalanceException


sd Sequence Diagram - View balance
: InterbankSubsystem
: InterbankSubsystemController
: InterbankBoundary
: Interbank
1: getBalance(card:CreditCard) : int
1.1: getBalance(card:CreditCard) : int
1.1.1: url = generateURL(data:Map<String,String>) : String
1.1.2: response = query(url:String) : String
1.1.2.1: processTransaction()
1.1.3: handleResponse(response:String) : Map<String,String>
throw InternalServerErrorException, InvalidCardException or NotEnoughBalanceException