Mini Test:

1. Demonstrate those above process/step for the two following requirements:
    a. System (Web/Mobile) allows the user to create their account.
    b. System (Web/Mobile) allows the users, who have account , buy items
    c. System (Web/Mobile) allows to manage their shopping cart(View item detail, add accessories, view feedback)
2. What are the main processes/steps of developing above System?
3. What details/diagrams will be used in each process/step?

1. Requirements: feasible

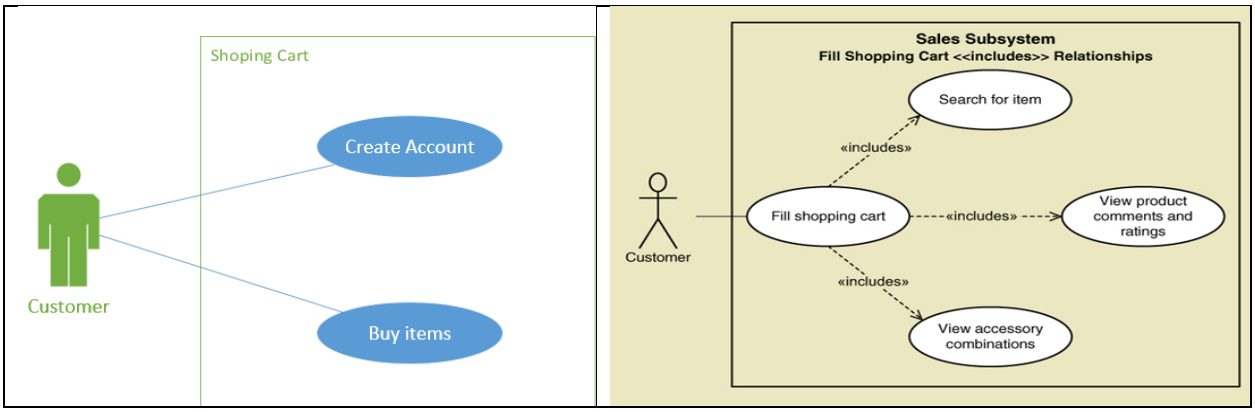
Requirements	Fun/Non- Function	Actor/User	Use Case (verb-Noun)
System allows the user to create their account.	Functional Req	Customer	Create Customer Account
System allows the users, who have account , buy items	Functional Req	Customer	Make order/Place Order / Buy Items/Purchase Items
System allows to manage their shopping cart	Functional Req	Customer	Fill shopping cart

(1000 reqs)

2. Discover and Understand the details (Analysis): Diagrams

- Information Systems: Process + Data

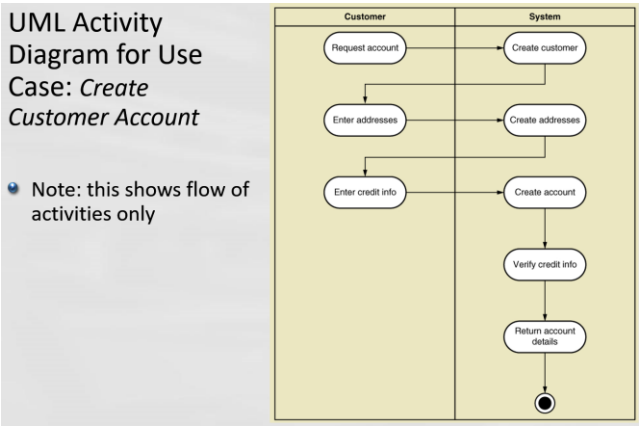
- Use Case Diagram:



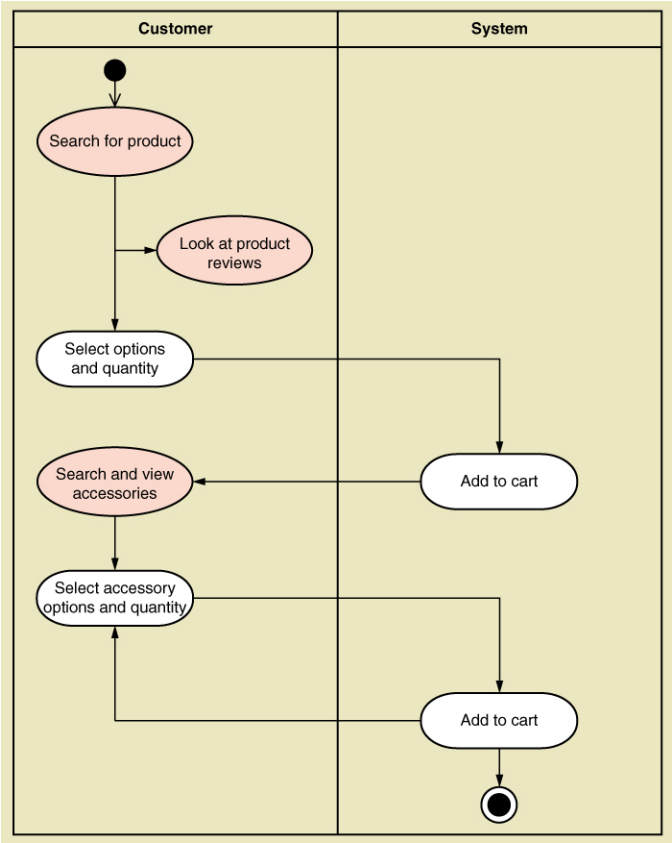
- **Process**: Workflow Diagram (Swim-lane Flowchart or UML Activity Diagram)

Activity Diagram :

1. Use case: Have 2 use cases -> 2 UML activity diagram: Process of Create Account / Process Buy Items

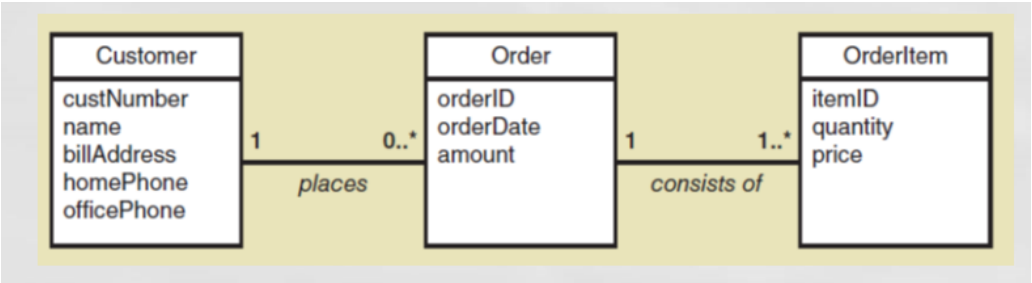


2. Sub-System: Overall workflow of a sub-system.
   U M L Activity Diagram for Use Case: Fill shopping cart this shows use case with <<includes>> relationship (***this use case diagram can be a sub-system***)

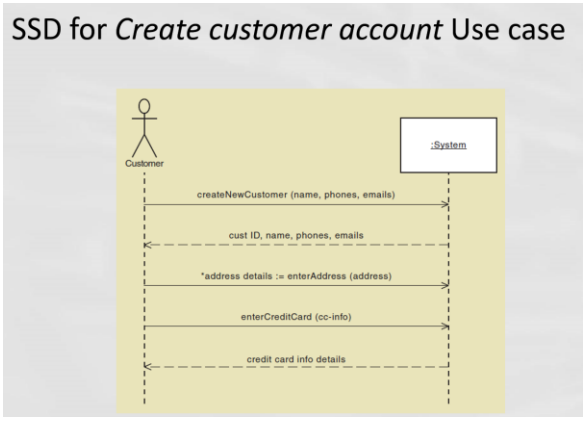


-**Data**:

1. Structure of system: Domain Class (Attributes) : chapter4



2. How actor/user interacts with the system: SSD : chapter 3/5

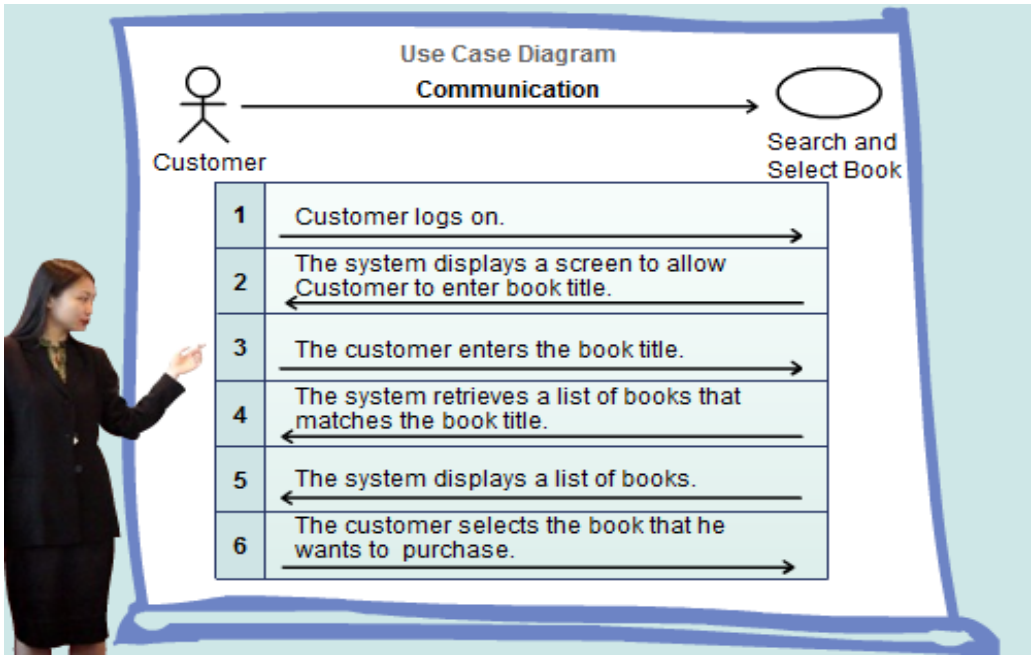


➔ State Machine Diagram: Pre-condition/Post Condition

Use Case Description

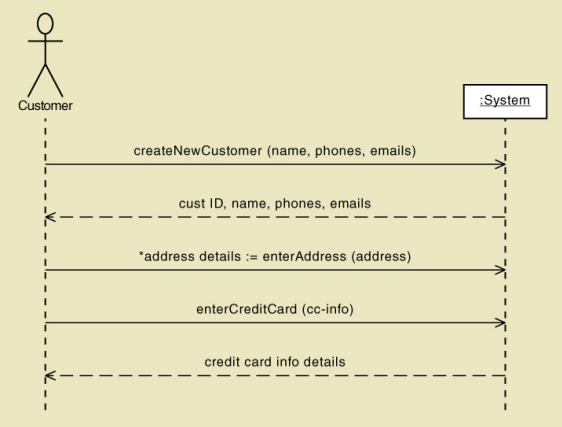
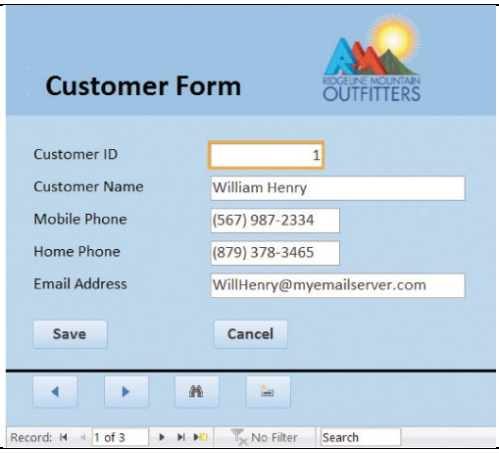
Use case name:	*Create customer account.*	
Scenario:	Create online customer account.	
Triggering event:	New customer wants to set up account online.	
Brief description:	Online customer creates customer account by entering basic information and then following up with one or more addresses and a credit or debit card.	
Actors:	Customer.	
Related use cases:	Might be invoked by the *Check out shopping cart* use case.	
Stakeholders:	Accounting, Marketing, Sales.	
Preconditions:	Customer Account subsystem must be available. Credit/debit authorization services must be available.	
Postconditions:	Customer must be created and saved. One or more Addresses must be created and saved. Credit/debit card information must be validated. Account must be created and saved. Address and Account must be associated with Customer.	
Flow of activities:	**Actor**	**System**
	1. Customer indicates desire to create customer account and enters basic customer information. 2. Customer enters one or more addresses. 3. Customer enters credit/debit card information.	1.1 System creates a new customer. 1.2 System prompts for customer addresses. 2.1 System creates addresses. 2.2 System prompts for credit/debit card. 3.1 System creates account. 3.2 System verifies authorization for credit/debit card. 3.3 System associates customer, address, and account. 3.4 System returns valid customer account details.
Exception conditions:	1.1 Basic customer data are incomplete. 2.1 The address isn't valid. 3.2 Credit/debit information isn't valid.	

[More Explain]: You need to understand the flow of activities, in some case, System Analyst also develop the communication diagram. Following is a use case communication diagram for a use case "Search and Select Book" (Dialog <-> sketch -> wireframe -> mockup-> prototype)

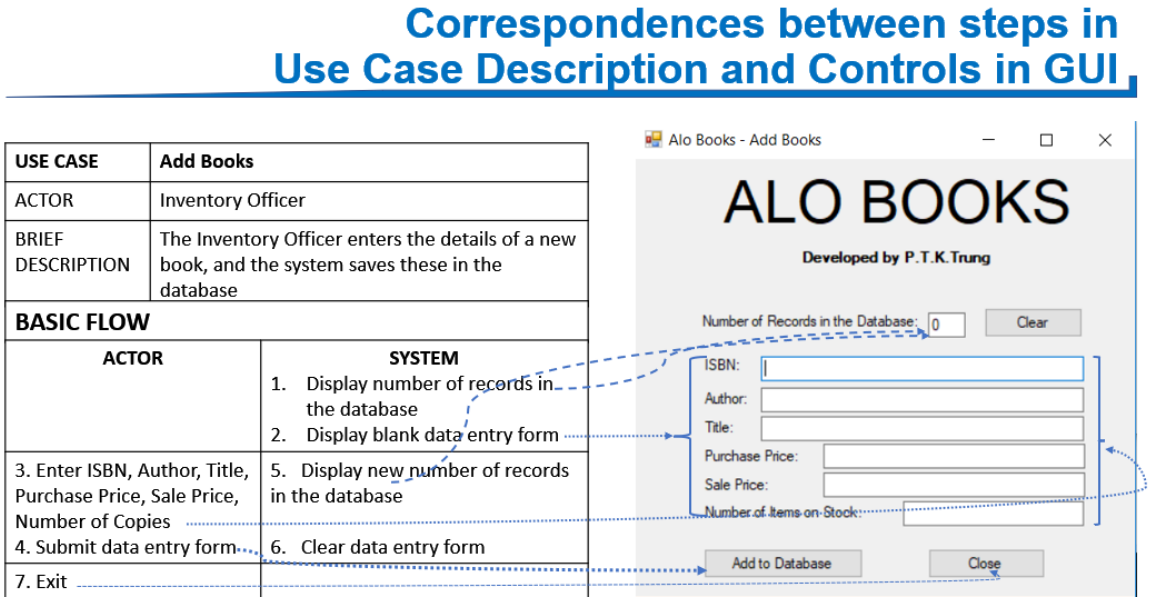


Design:

- UI: User Interface: Transfer SSD -> GUI

SSD: System Sequence Diagram	UI
	

Other example: (*In a real world, we may need a Use Case Description then developer can develop a GUI for use case directly*)



USE CASE	Add Books
ACTOR	Inventory Officer
BRIEF DESCRIPTION	The Inventory Officer enters the details of a new book, and the system saves these in the database
BASIC FLOW	
ACTOR	**SYSTEM**
	1. Display number of records in the database 2. Display blank data entry form
3. Enter ISBN, Author, Title, Purchase Price, Sale Price, Number of Copies 4. Submit data entry form	5. Display new number of records in the database 6. Clear data entry form
7. Exit	

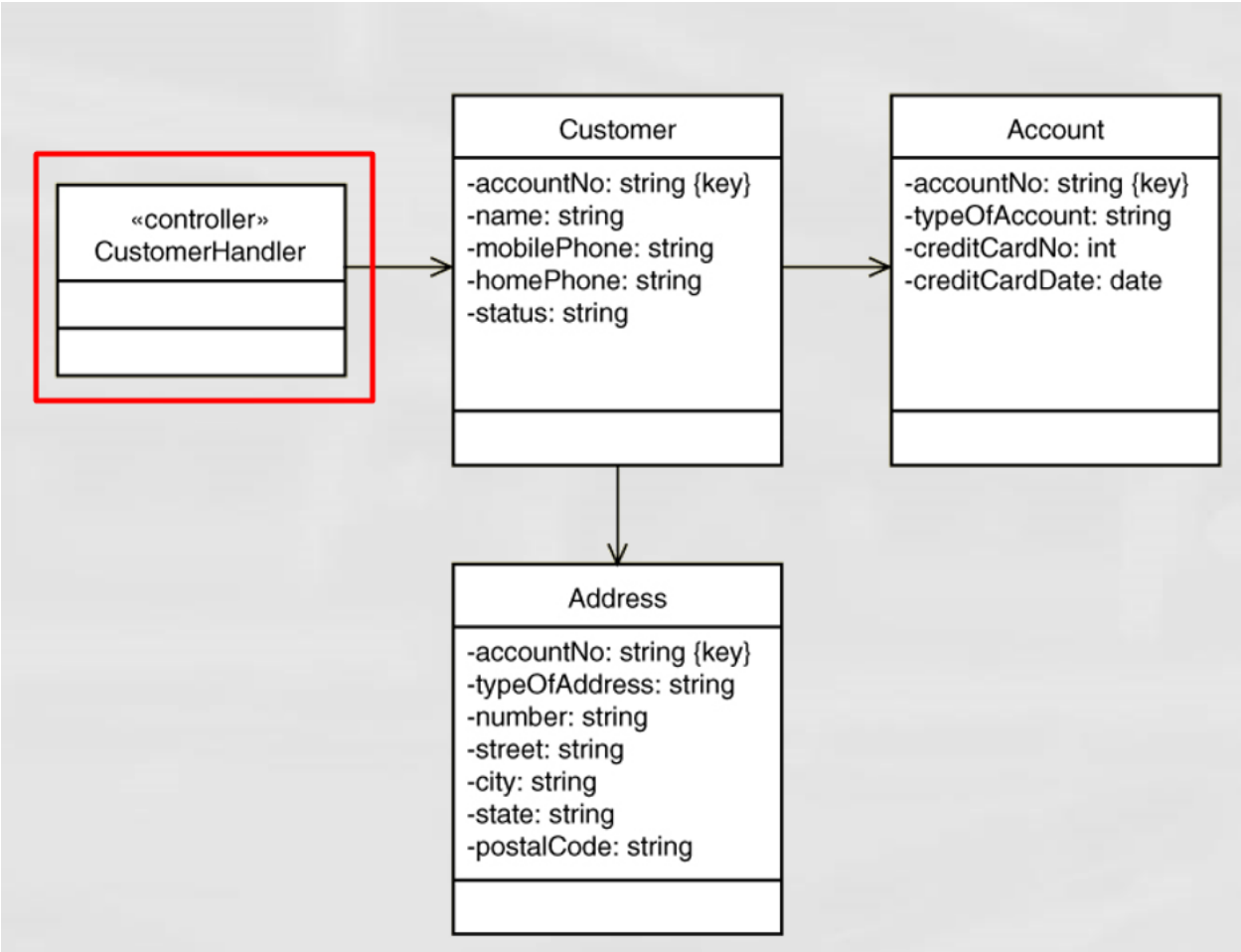
- Architecture Design
  Refer to chapter 7: Architecture Design (3 tiers: UI, Controller, DB)
- Database Design:
  Refer to chapter 7: Transfer Domain Class -> Relational Schema

-Design Class Diagram (not required in your individual assignment but optional for extra point)

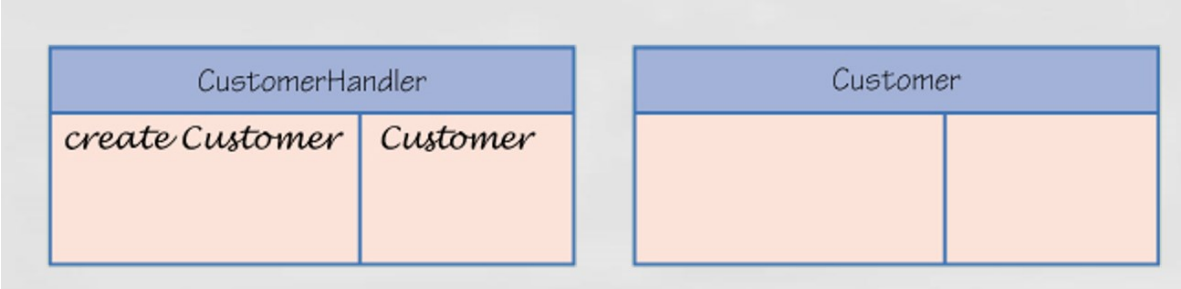
Completely developing “design class diagram”

Refer to chapter 12,13

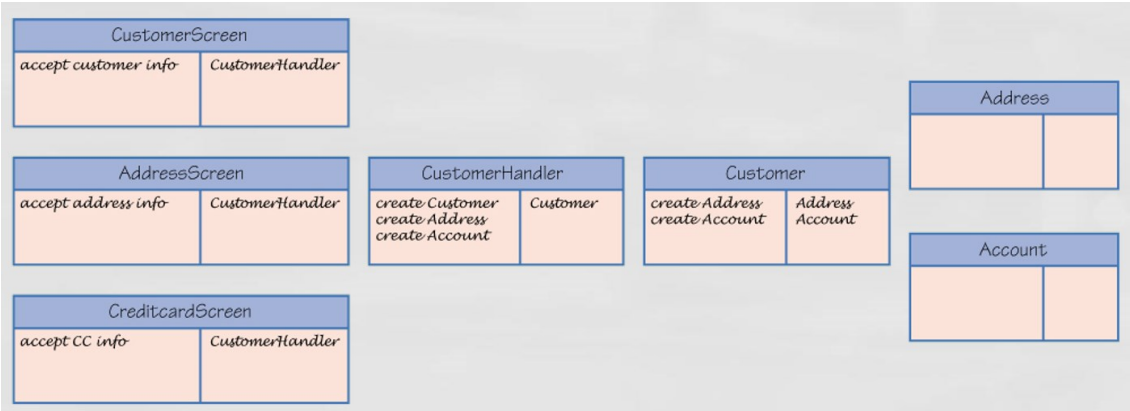
1. Domain Design Class



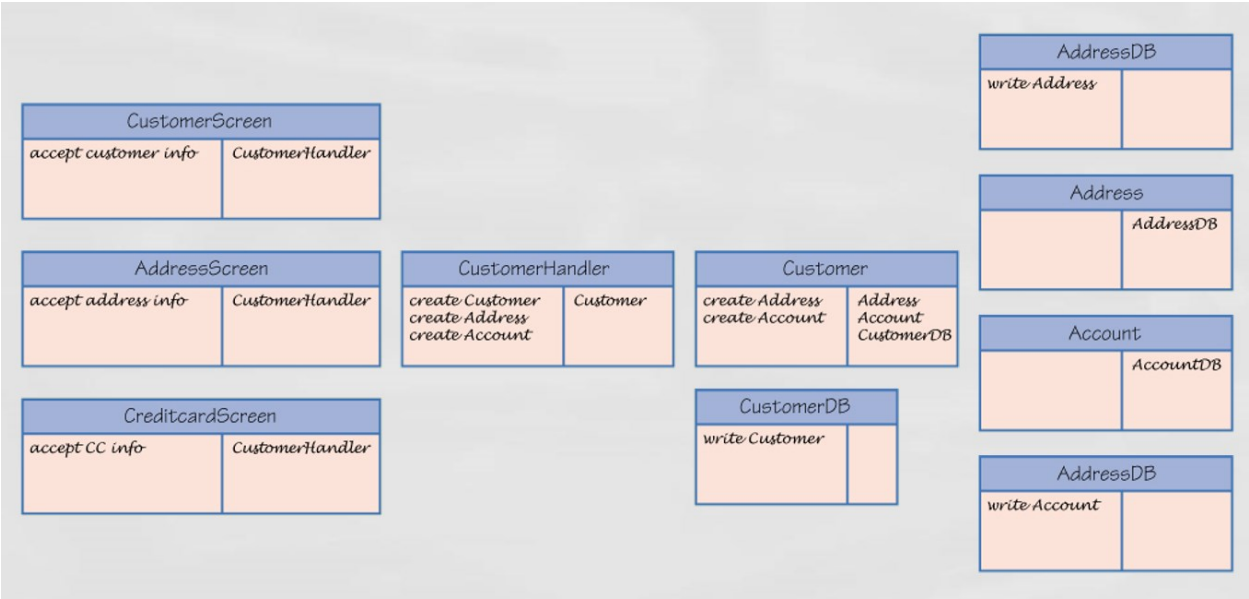
2. Controller



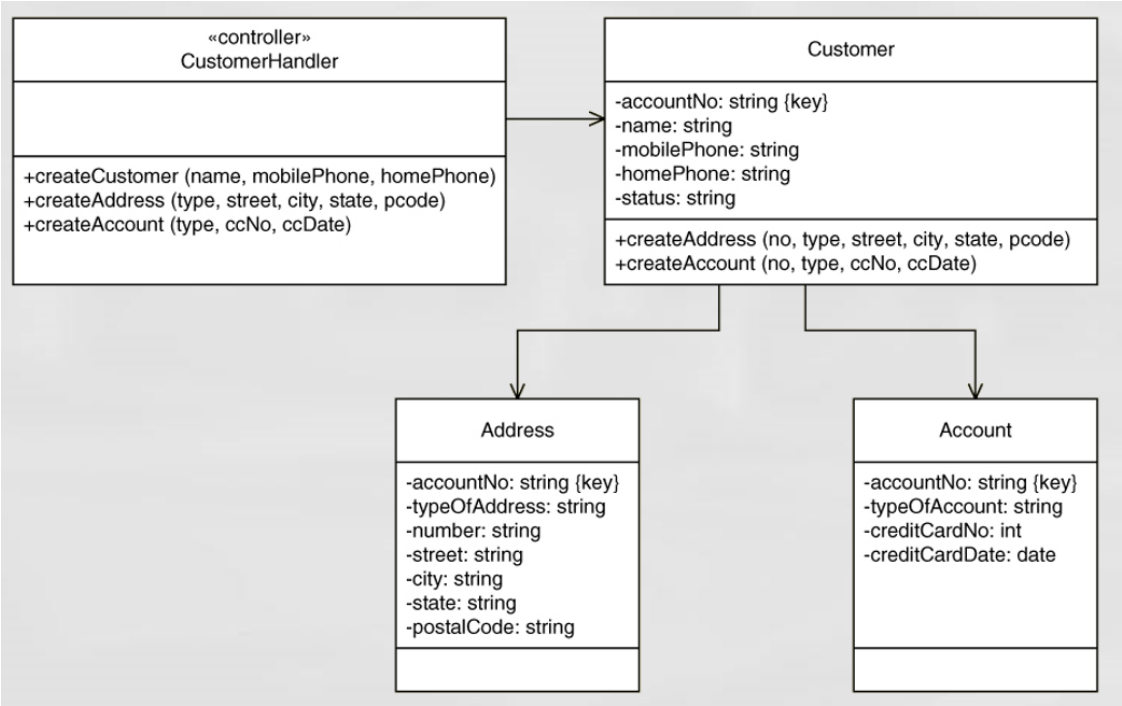
3. UI



4. Data Access



2. Design Class



3. Code, Test

Interface createCustomer{

+/-/# attribute

Public createCustomer()

}

Implementation:

Deployment /Component Diagram

Email: tg_phamthaikytrung@tdtu.edu.vn if you are students of TDTU.