INFORMATION SYSTEM ANALYSIS AND DESIGN – 2024

PROJECT

BUILD PART OF SPECIFICATION, DESIGN DOCUMENT AND PROGRAMMING

Each student must take part in all activities of development process

- Requirement determination and analysis
- Design overall and detail
- Programming

Tasks	Detail tasks	Students	Due date
Req. Deter. & Analysis			**21/10—27/10**
Design			**04/11—10/11**
Programming			**18/11—25/11**

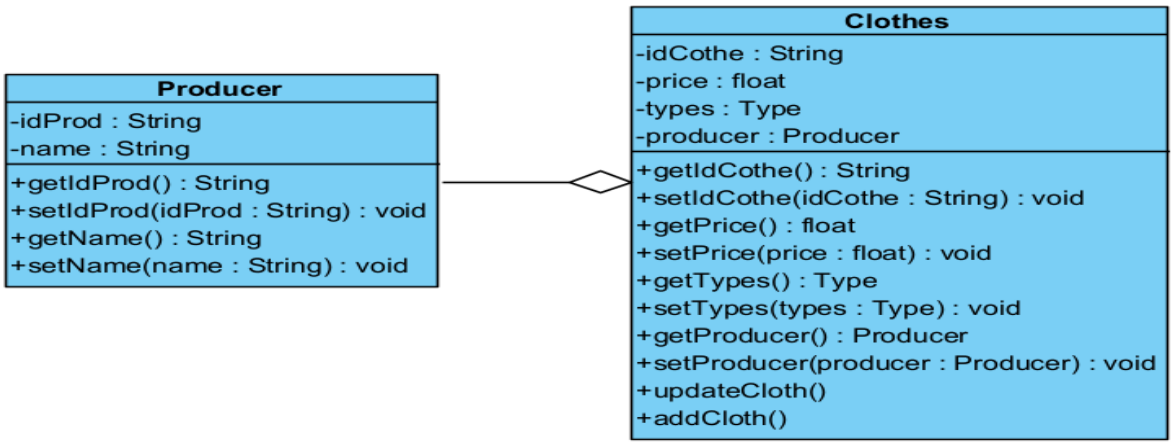
Develop an e-commerce site, which may be users by types of customers, types of staff, managers. This site provides services of purchasing *books, clothes, mobile phone, laptop, shoes, electronic devices*. Your tasks:

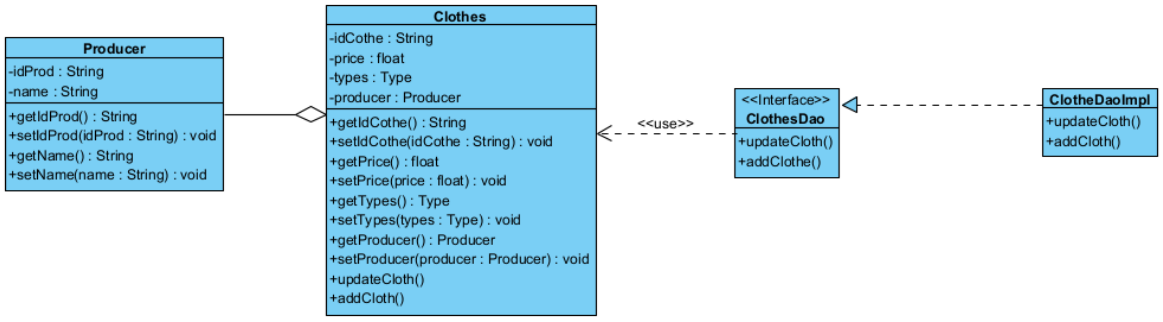
1. Determine requirements
   - Define use cases and draw use case diagram in two levels: General level (>=20 use cases) and Level 1 (detail)
   - Write 30 scenarios and 30 user stories + acceptance criteria
   - Draw 5 activity/swimlane diagrams + 5 sequence diagrams for the processes executed by customer and staff:
     - Create cart, checkout, select ship (drone), select pay and order, give comments & feed back of items
     - Process order: alert order (automatically), check order, send invoice/feedback to customer, send requirements to ship department and repository
     - Process payment
     - Process shipment (using drone)
2. Requirement Analysis
   - Write 50 CRC Cards
   - Draw a class diagram (analysis) with 50 classes
   - Construct data dictionary
   - Create data model
3. Design
   - Construct a class diagram (design)
     - Validate the class diagram in analysis (Name, attributes, methods, relationship...)

    - Add attributes, add methods, change multiplicity, change relationship
    - Change association classes into suitable relationships (aggregation, composition)
- Using MVC model to design the e-commerce system
    - Using DAO pattern to classes
    - Using package notion in UML to group classes into packages and layers with the model MVC: Model layer, Controller model, View model
- Construct database
    - Validate data model: check fields, key, connection multilicity...
    - Select a model for inheritance
    - Generate database with
- Design interface (user interface and system interface)
- Design forms such as bill, report....

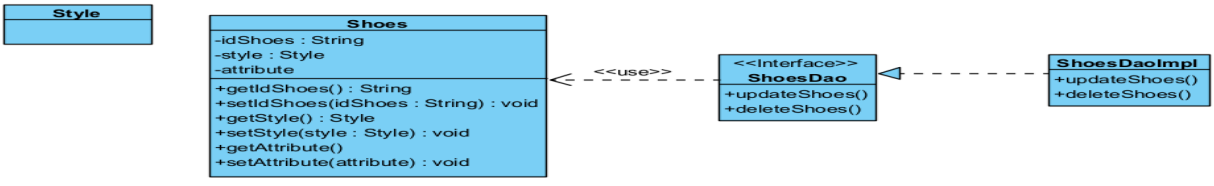
4. Programming and Integration
    - Using Java Spring (Boot), JSP, CSS, NoteJS....
    - Developing/utilizing techniques in deep learning to build modules search, analyze opinions/comments, recommender items, generate plots of analyzing customer's tendency….

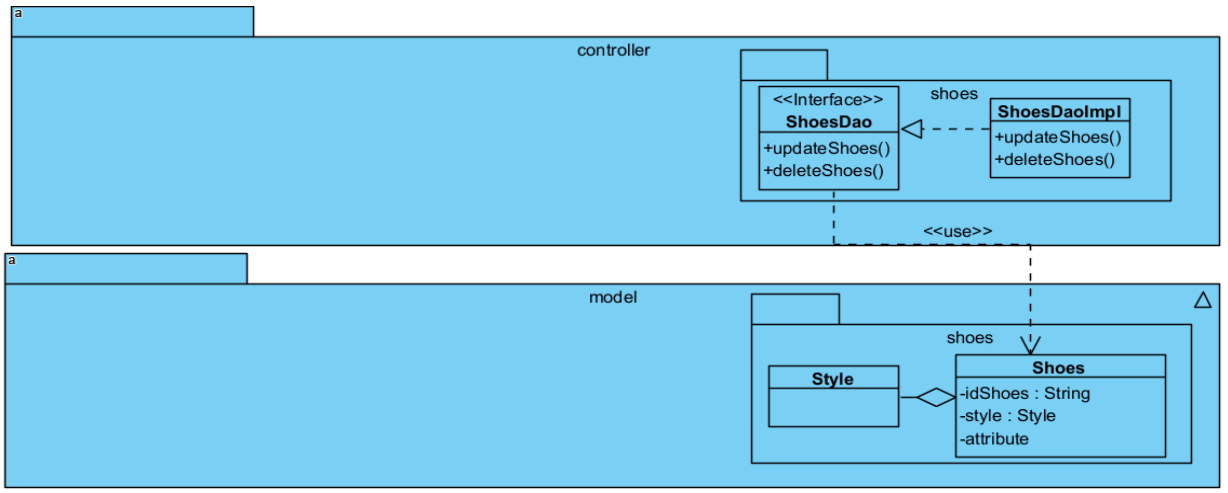
REFERENCES


Producer
-idProd : String
-name : String
+getIdProd() : String
+setIdProd(idProd : String) : void
+getName() : String
+setName(name : String) : void
Clothes
-idCothe : String
-price : float
-types : Type
-producer : Producer
+getIdCothe() : String
+setIdCothe(idCothe : String) : void
+getPrice() : float
+setPrice(price : float) : void
+getTypes() : Type
+setTypes(types : Type) : void
+getProducer() : Producer
+setProducer(producer : Producer) : void
+updateCloth()
+addCloth()


Producer
-idProd : String
-name : String
+getIdProd() : String
+setIdProd(idProd : String) : void
+getName() : String
+setName(name : String) : void
Clothes
-idCothe : String
-price : float
-types : Type
-producer : Producer
+getIdCothe() : String
+setIdCothe(idCothe : String) : void
+getPrice() : float
+setPrice(price : float) : void
+getTypes() : Type
+setTypes(types : Type) : void
+getProducer() : Producer
+setProducer(producer : Producer) : void
+updateCloth()
+addCloth()
<<use>>
<<Interface>>
ClothesDao
+updateCloth()
+addClothe()
ClotheDaoImpl
+updateCloth()
+addCloth()

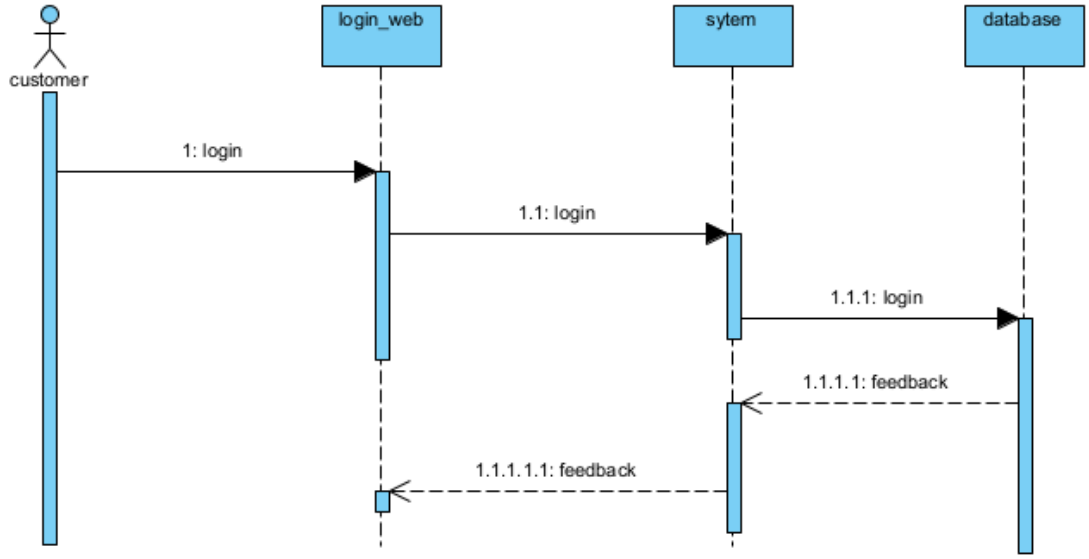

Style
Shoes
-idShoes : String
-style : Style
-attribute
+getIdShoes() : String
+setIdShoes(idShoes : String) : void
+getStyle() : Style
+setStyle(style : Style) : void
+getAttribute()
+setAttribute(attribute) : void
+updateShoes()
+deleteShoes()


Style
Shoes
-idShoes : String
-style : Style
-attribute
+getIdShoes() : String
+setIdShoes(idShoes : String) : void
+getStyle() : Style
+setStyle(style : Style) : void
+getAttribute()
+setAttribute(attribute) : void
<<use>>
<<Interface>>
ShoesDao
+updateShoes()
+deleteShoes()
ShoesDaoImpl
+updateShoes()
+deleteShoes()

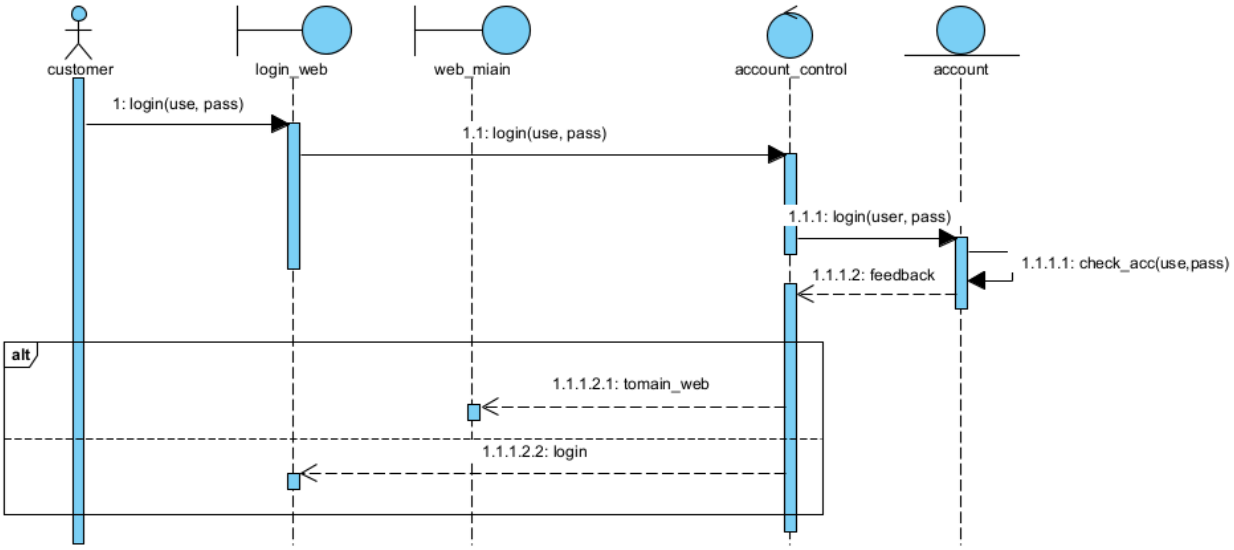


SEQUENCE DIAGRAM

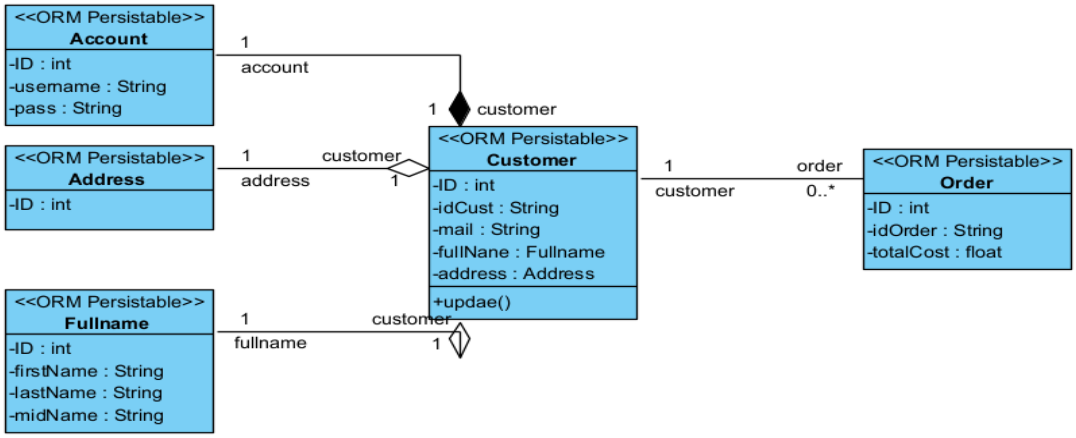
Initial diagram Sequence in the analysis phase



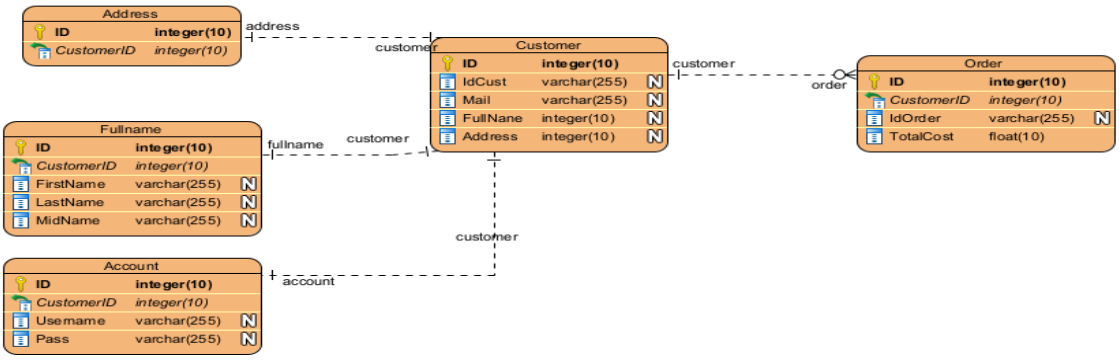
Sequence Diagram in design phase



BUILDING DATA MODEL

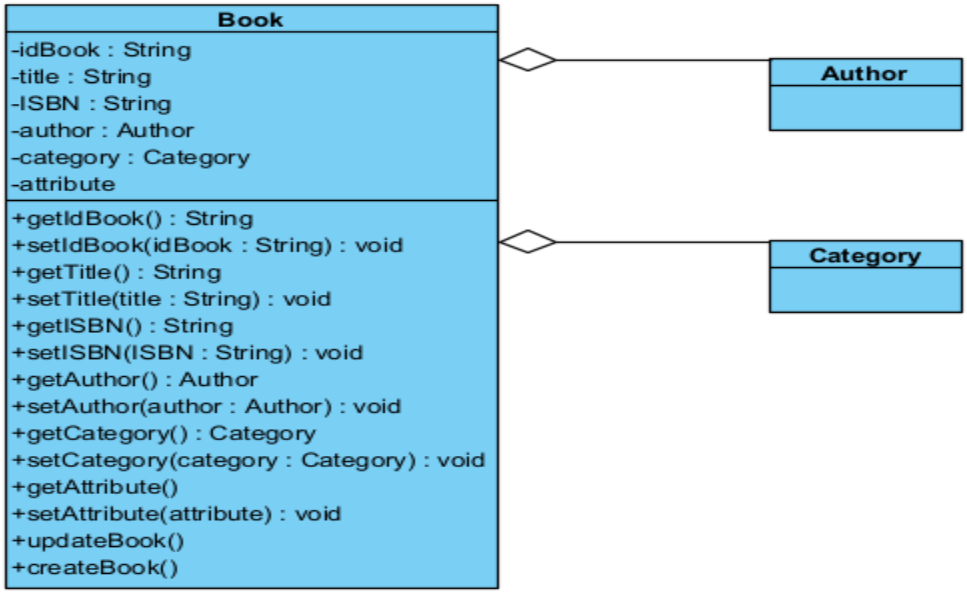


DATA MODEL

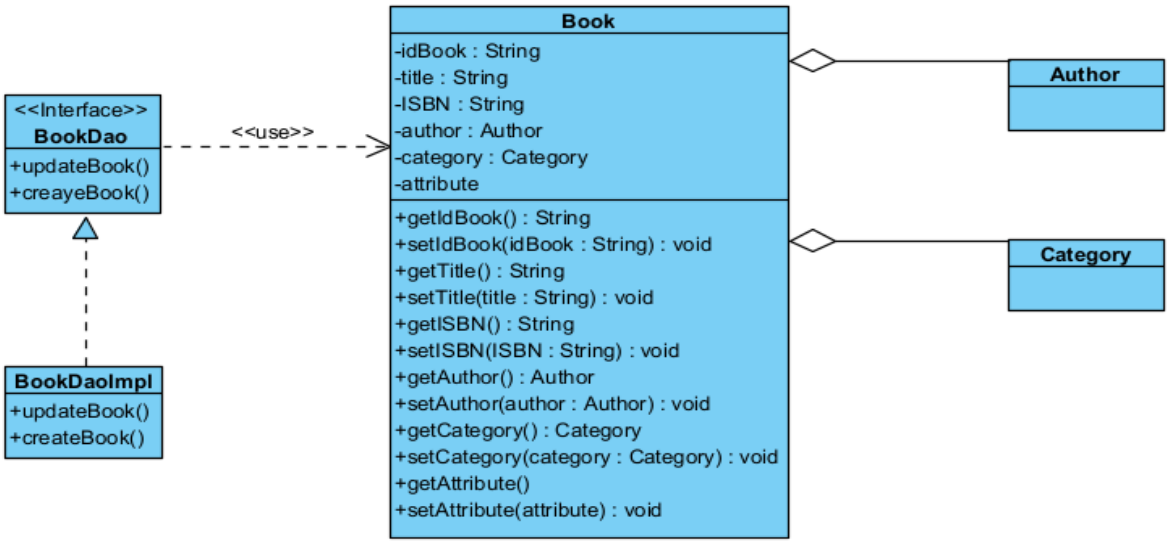


DESIGN STEPS

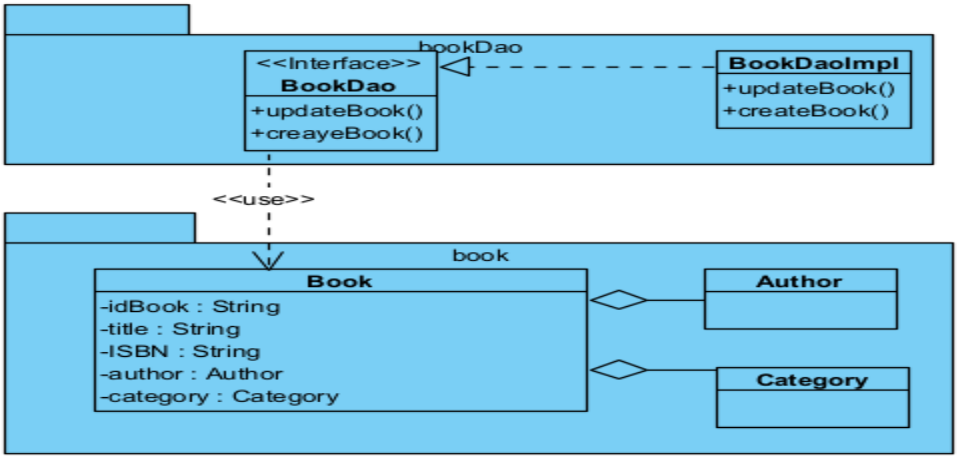
STEP 1: VALIDATE CLASS



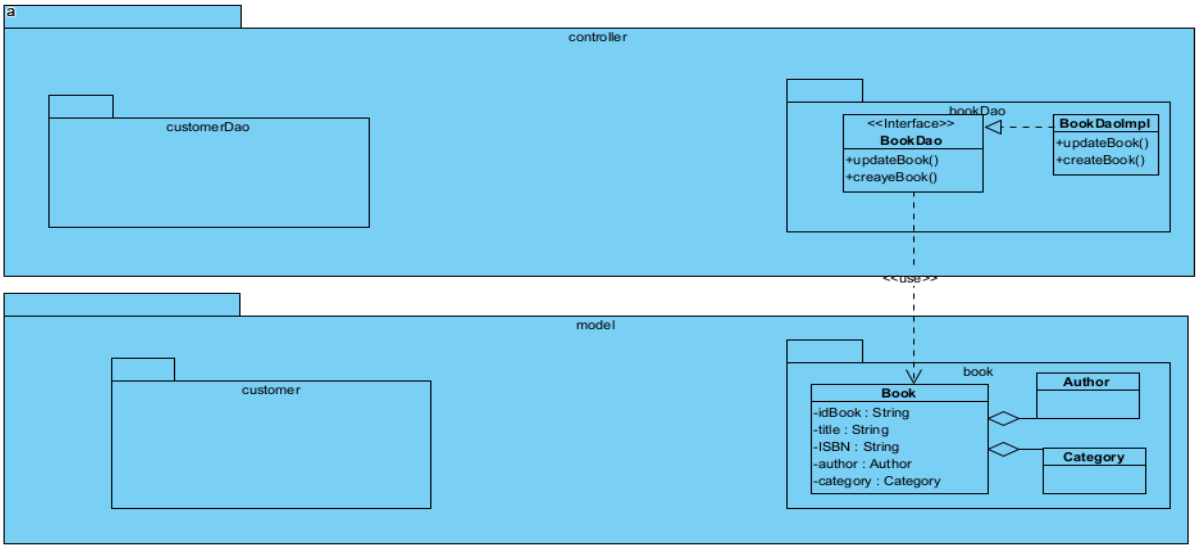
STEP 2 CREATE DAO



STEP 3: PACKAGE



STEP 4: Layer



EXAMPLE OF THE LAST DESIGN