

Ex.No. 1 Ecommerce Website

Date: 12/05/2023

Aim :

Analysis and Identification of suitable process model for “Ecommerce Website”

Description:

An ecommerce website is an online platform that allows businesses to sell products or services to customers over the internet. It serves as a virtual storefront, providing customers with a digital shopping experience where they can browse, select, and purchase items from the comfort of their own homes.

A well-designed ecommerce website typically includes several key features. Product catalog pages display detailed information about the available products, including images, descriptions, pricing, and customer reviews. Shopping carts enable customers to add items they wish to purchase and proceed to the checkout process. Secure payment gateways ensure the safe and convenient processing of online transactions.

To enhance the user experience, ecommerce websites often incorporate search functionality, filters, and sorting options to help customers find products efficiently. Personalized recommendations and related product suggestions may also be included to assist customers in discovering additional items of interest.

Process Models:

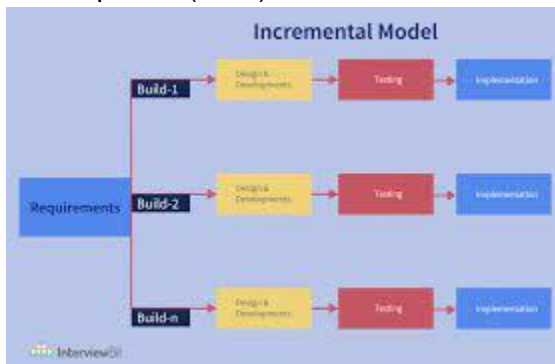
WATERFALL MODEL:

The waterfall model is a sequential and linear approach to software development. It follows a systematic and structured progression through distinct phases, with each phase acting as a foundation for the next. The waterfall model typically consists of the following phases: requirements gathering, system design, implementation, testing, deployment, and maintenance. In this model, each phase has clearly defined inputs, outputs, and objectives. The progress flows in a downward direction, similar to a waterfall, and once a phase is completed, there is minimal or no going back to previous stages. The waterfall model emphasizes a comprehensive upfront planning and documentation, which can help establish a clear understanding of the project requirements and minimize potential risks. However, it can be less flexible in accommodating changes and may lead to delayed feedback from users until later stages.



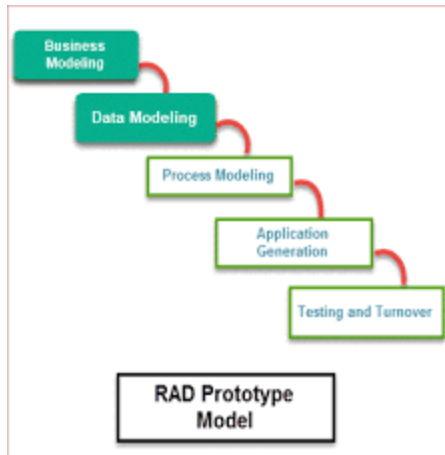
INCREMENTAL MODEL:

The incremental process model is an iterative approach to software development that focuses on delivering the software in small, incremental portions. Each increment adds new functionality to the system, building upon the previous increments. This model allows for early feedback and provides opportunities for customer involvement throughout the development process. Two subtypes of the incremental model are the Incremental Model and the Rapid Application Development (RAD) Model.



The Incremental Model breaks down the development process into multiple increments or iterations, with each iteration delivering a fully functional subset of the software. The requirements are divided into modules, and each module is developed and tested independently. This approach allows for early delivery of working software and facilitates easier management of changing requirements. It also provides the opportunity to incorporate user feedback and make necessary adjustments in subsequent increments.

The Rapid Application Development (RAD) Model is a variation of the incremental model that focuses on speed and efficiency. It emphasizes the use of prototyping and involves an intense and collaborative development process. RAD aims to deliver high-quality software in a shorter time frame by using techniques such as workshops, iterative prototyping, and time-boxing. RAD places a strong emphasis on active user involvement and feedback to ensure that the final product meets user expectations.



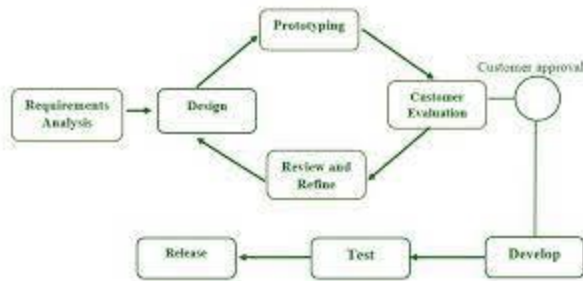
EVOLUTIONARY MODEL:

The evolutionary process model is an iterative and incremental approach to software development that allows for continuous refinement and evolution of the software product. It recognizes that software requirements and solutions may change over time and accommodates these changes through iterative development cycles. The evolutionary process model focuses on delivering a basic working version of the software early on and then gradually improving it through subsequent iterations. This model encourages user involvement and feedback at each iteration, enabling stakeholders to provide input and make adjustments as the project progresses. Two common subtypes of the evolutionary process model are the Prototyping Model and the Spiral Model.



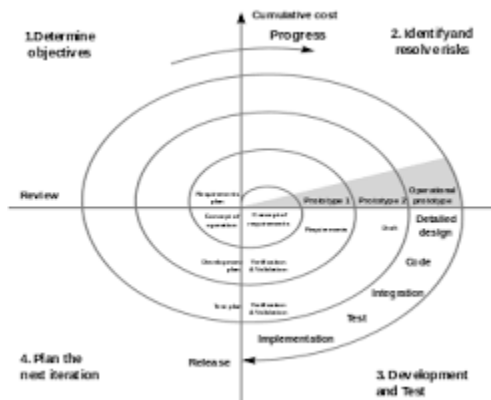
Prototyping model:

The Prototyping Model emphasizes the development of a quick, simplified version of the software to gather user feedback and validate requirements. The prototype serves as a tangible representation of the desired system and helps stakeholders visualize the end product. Based on user feedback, the prototype is refined and iterated upon until it aligns with the user's expectations. This model promotes effective communication and understanding between the development team and the users, reducing the risk of misunderstandings or misinterpretations.



Spiral model:

The Spiral Model combines elements of both the waterfall model and the iterative approach. It involves multiple iterations or cycles, each consisting of four main activities: planning, risk analysis, development and evaluation. The Spiral Model incorporates risk management as a core component, identifying and addressing potential risks at each cycle. It allows for progressive development and frequent evaluation of the software, providing opportunities to adapt to changing requirements and evolving risks. This model is particularly suitable for complex projects with high levels of uncertainty and varying priorities.



AGILE MODEL:

Agile models are a set of methodologies and frameworks that prioritize flexibility, collaboration, and iterative development in software engineering. The Agile approach emphasizes adaptive planning, continuous feedback, and delivering working software in short, incremental iterations known as sprints.



Scrum is one of the most widely used Agile models. It divides the project into time-boxed iterations called sprints, usually lasting 1-4 weeks. Scrum teams self-organize and collaborate to

deliver small, valuable increments of the software at the end of each sprint. Daily stand-up meetings, backlog refinement, sprint planning, and retrospective sessions are integral to Scrum.

Extreme Programming (XP) is an Agile model that emphasizes close collaboration, simplicity, and high-quality software. XP employs practices like test-driven development (TDD), pair programming, continuous integration, and frequent customer involvement. It promotes rapid feedback, adaptability, and constant improvement.

Agile models offer benefits such as increased customer satisfaction, faster delivery, and the ability to accommodate changing requirements. They encourage transparency, collaboration, and self-organizing teams. However, Agile models require a cultural shift, effective communication, and skilled practitioners to achieve their full potential. Regular reflection and adaptation are essential to continuously improve the development process and deliver valuable software.

Output:

Identification of suitable process model for your application

For the development of an ecommerce website, the Agile model, particularly Scrum, would be a suitable choice. Here are the reasons:

- 1. Flexibility and Adaptability:** The Agile model allows for flexibility in accommodating changing requirements. With an ecommerce website, market trends, customer preferences, and business goals may evolve over time. Agile methodologies like Scrum enable iterative development, allowing the website to be continuously refined and adjusted based on customer feedback and emerging needs.
- 2. Time-to-Market:** The ecommerce industry is highly competitive and demands rapid delivery of new features and updates. Agile models emphasize delivering working software in short iterations, enabling quicker time-to-market. This ensures that the website can be launched with essential functionalities, and subsequent iterations can add additional features incrementally.
- 3. Customer-Centric Approach:** Agile models, including Scrum, prioritize customer involvement and feedback. This is crucial for an ecommerce website as it directly caters to customer needs and preferences. Regular customer engagement throughout the development process allows for early identification of issues, refinement of features, and validation of user experience.
- 4. Collaboration and Communication:** Agile methodologies foster collaboration among cross-functional teams, including developers, designers, and stakeholders. This facilitates effective communication and alignment of objectives. For an ecommerce website, this collaboration is

essential for seamless integration of features, user interface design, and efficient management of product catalog and inventory.

5. Continuous Improvement: Agile models promote a culture of continuous improvement and learning. With an ecommerce website, there is an ongoing need to enhance the user experience, optimize conversion rates, and adapt to changing market dynamics. Agile practices enable regular retrospectives and the ability to adapt the development approach based on lessons learned and emerging market trends.

Overall, the Agile model, specifically Scrum, is a suitable choice for ecommerce website development due to its flexibility, customer-centric approach, quick time-to-market, collaboration, and continuous improvement focus. It ensures the website can be developed iteratively, meeting evolving customer needs while allowing for rapid delivery and ongoing optimization.

Result : Thus the analysis of various process models and identification of the suitable process model for the project “ ecommerce website” is successfully done. The identified process model is “Agile Model“

Ex.No. 2

Title – As per Your lab listing

Date: 02 May 2023

Aim :

To draw Gantt Chart for “ecommerce website”

Description:

About Gantt Chart

Output:

Result : Thus the Gantt chart is designed for planning the timeline of various activities for the project “ Your application”

Name of Student:

Vaibhav Singh

Reg. No.:21BCE5616

Ex. No: 03

Date: 03/02/23

REQUIREMENTS MODELING USING ER
DIAGRAM

Aim: To prepare an Entity Relationship (ER) Diagram that represents relationships among different entities in a database.

Tools Used: Star UML software

Description:

An **entity relationship diagram (ER)**, also known as an entity relationship model, is a graphical representation that depicts relationships among people, objects, places, concepts or events within an information technology system. An ERD uses data modeling techniques that can help define business processes and serve as the foundation for a relational database.

There are five basic components of an entity relationship diagram. The components include:

1. **Entities**, which are objects or concepts that can have data stored about them. Entities refer to tables used in databases.
2. **Attributes**, which are properties or characteristics of entities. An ERD attribute can be denoted as a primary key, which identifies a unique attribute, or a foreign key, which can be assigned to multiple attributes.
3. The **relationships** between and among those entities.
4. **Actions**, which describe how entities share information in the database.
5. **Connecting lines**

Relations:

1. Categories:

Category_id primary key int

Category_name varchar

Category_type varchar

2. Shopping_Order

Order_id primary key

customer_id foreign key

Date {date}

3. Deliveries

deliveries_id primary key

Customer_id foreign key

Date {date}

4. Products

Product_id primary key

Category_id foreign key

Product_name

5. Customers

Customer_id primary key

Name char

contact_add varchar

Address varchar

6. Transaction reports

Report_id primary key

Customer_id foreign key

Order_id foreign key

Product_id foreign key

payment_id foreign key

7. Seller

Seller_id primary key

Product_id foreign key

Name varchar

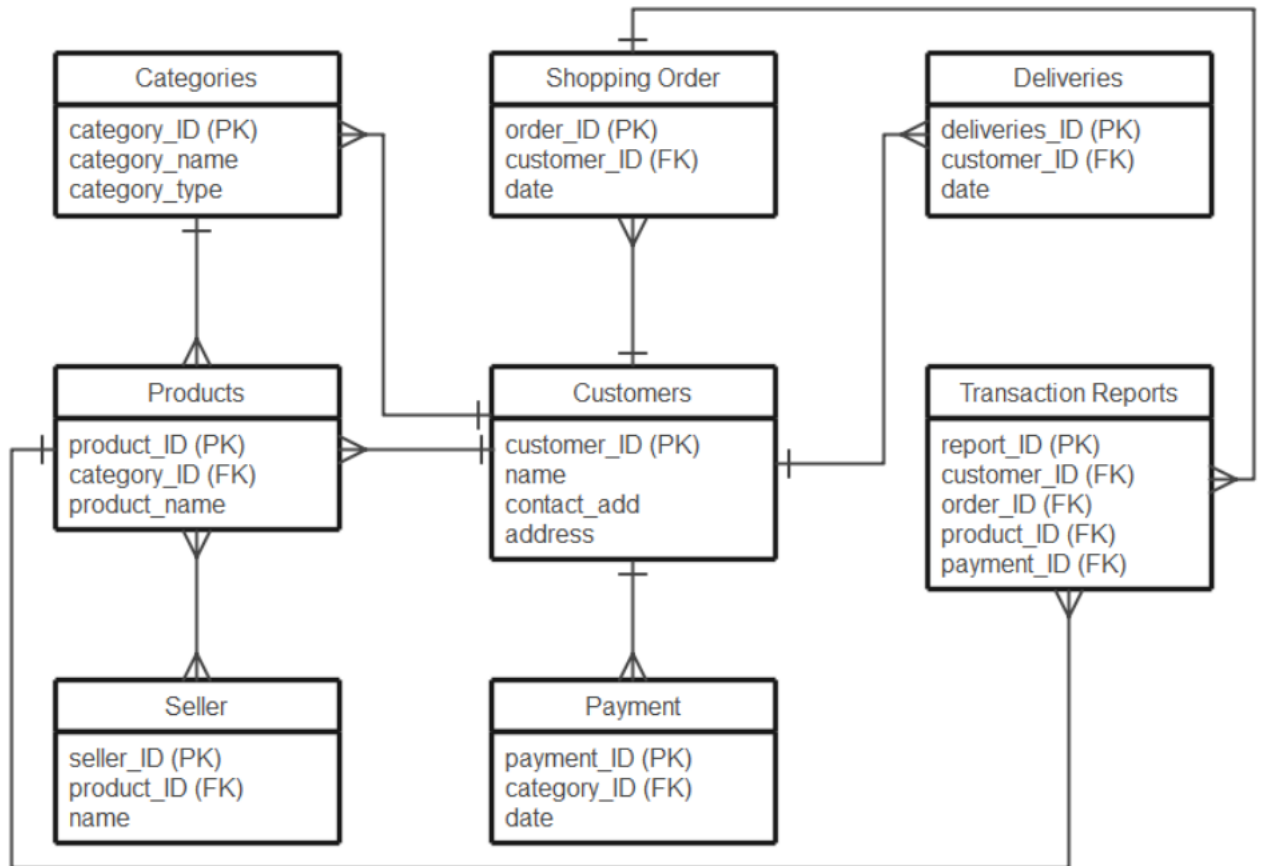
8. Payment

Payment_id primary key

Category_id foreign key

Date varchar

Output Screenshot:



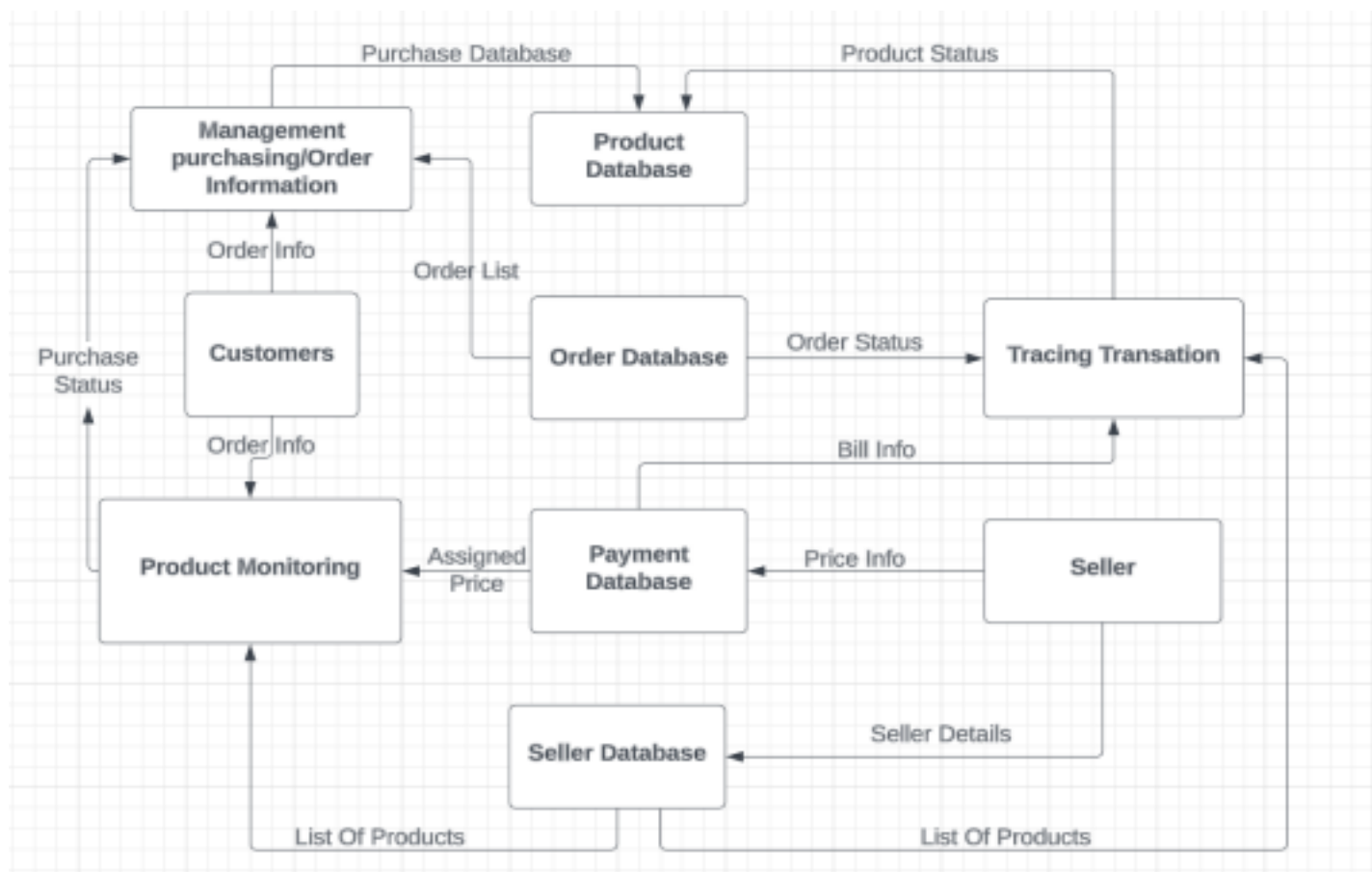
Result:

Thus, the ER Diagram was prepared for the application e commerce website

Data Flow Diagram For Ecommerce Website

Vaibhav Singh (21BCE5616)

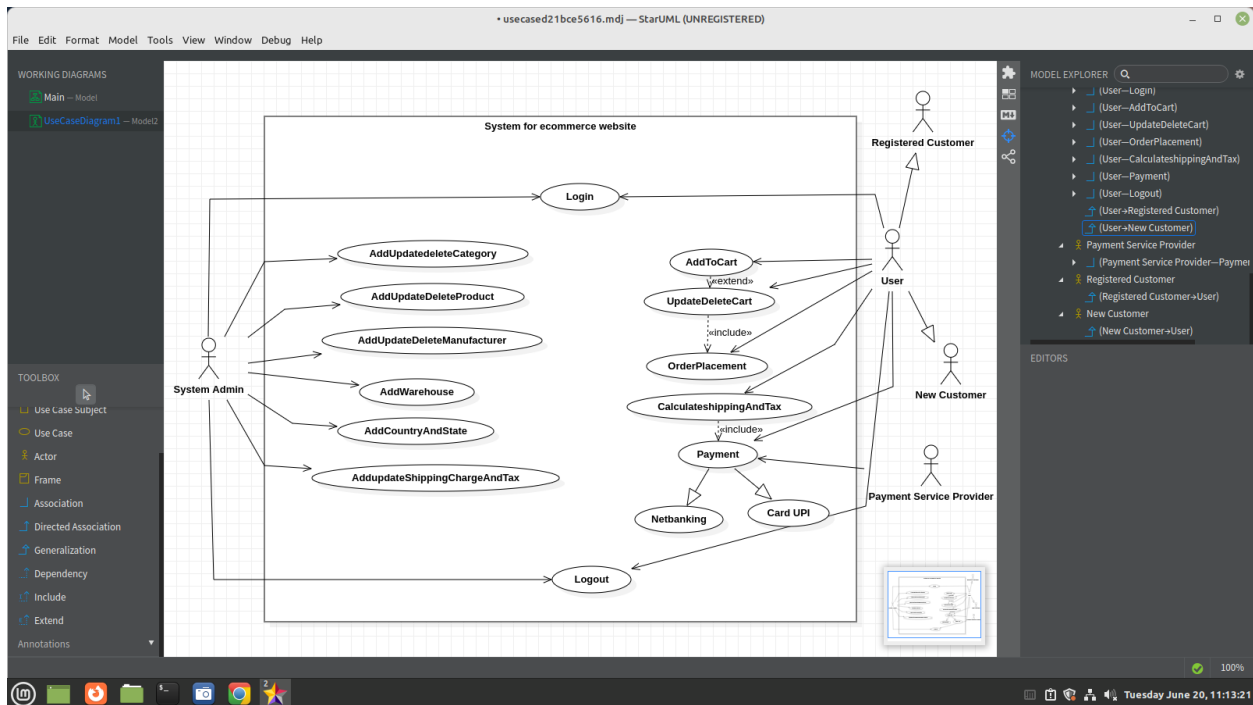
Data Flow Diagram (DFD) is a visual representation of a software system's flow of data. It is a modeling technique used in software engineering to analyze, design, and represent the flow of data and processes within a system.



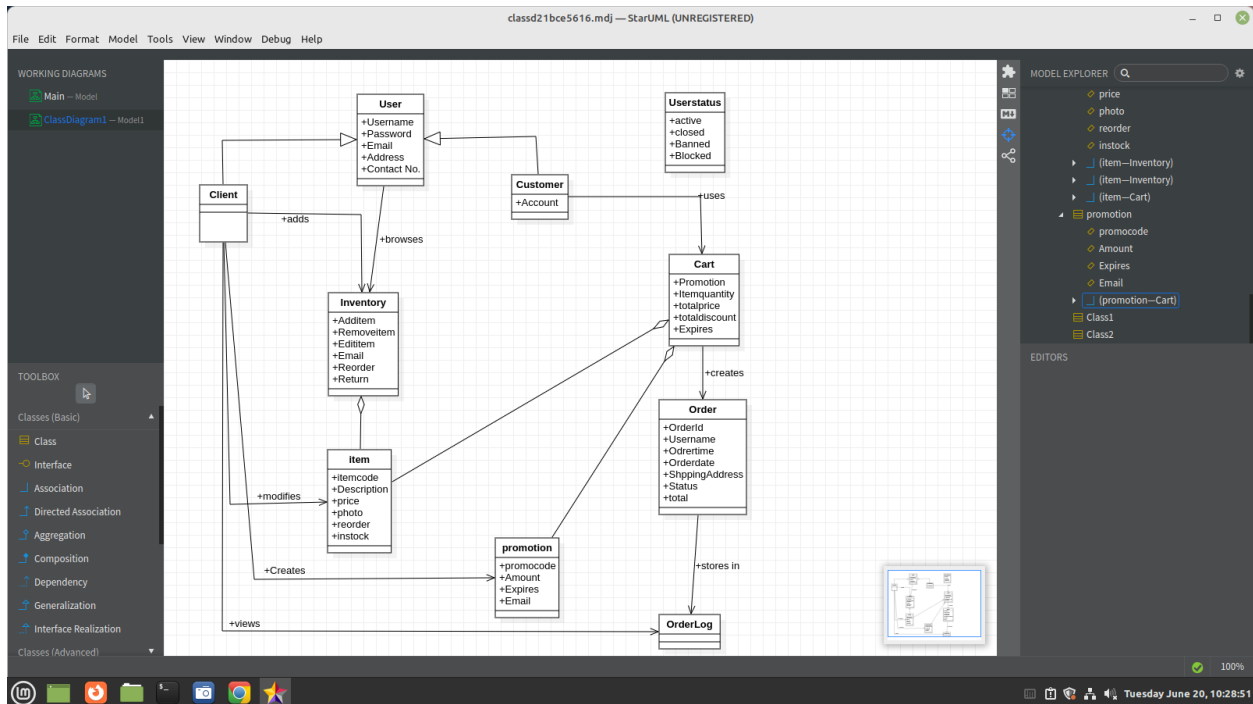
NAME: VAIBHAV SINGH
REG. : 21BCE5616

SOFTWARE ENGINEERING LAB (EXP - 5)

USE CASE DIAGRAM:



CLASS DIAGRAM:



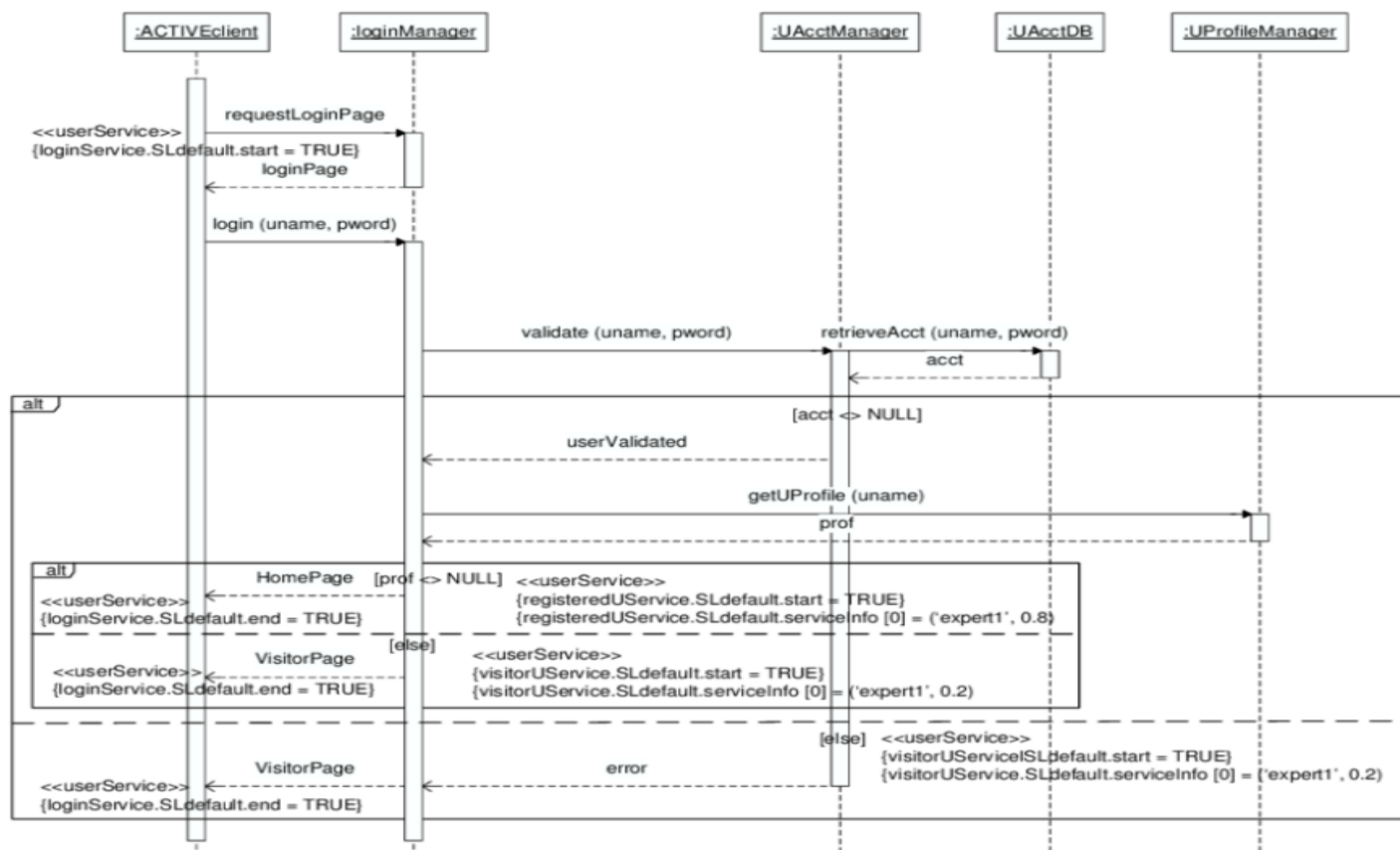
Software Lab Assignment

Vaibhav Singh

21BCE5616

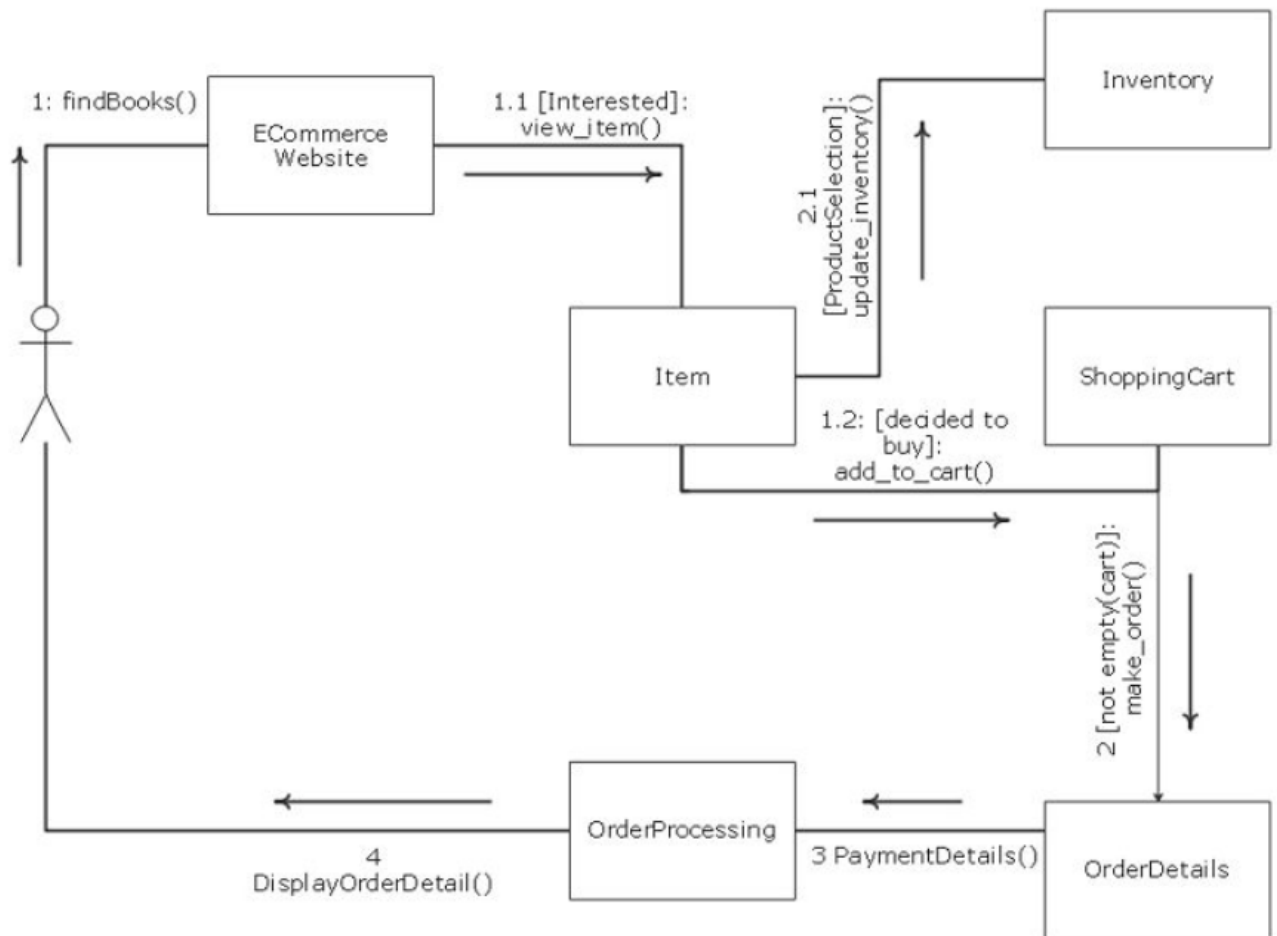
Sequence diagram

A sequence diagram is a type of UML (Unified Modeling Language) diagram used in software engineering to model the interactions and relationships between various objects or actors in a system. It shows the sequence of messages exchanged between objects over time, illustrating how they interact with each other to accomplish a particular task or function.



Collaboration Diagram:

A collaboration diagram, also known as a communication diagram, is a type of UML (Unified Modeling Language) diagram used to visualize the interactions and relationships between objects or actors in a system. It shows the flow of messages or information between objects and can be used to model both synchronous and asynchronous communication.



Ex.No. 7

eCommerce Website

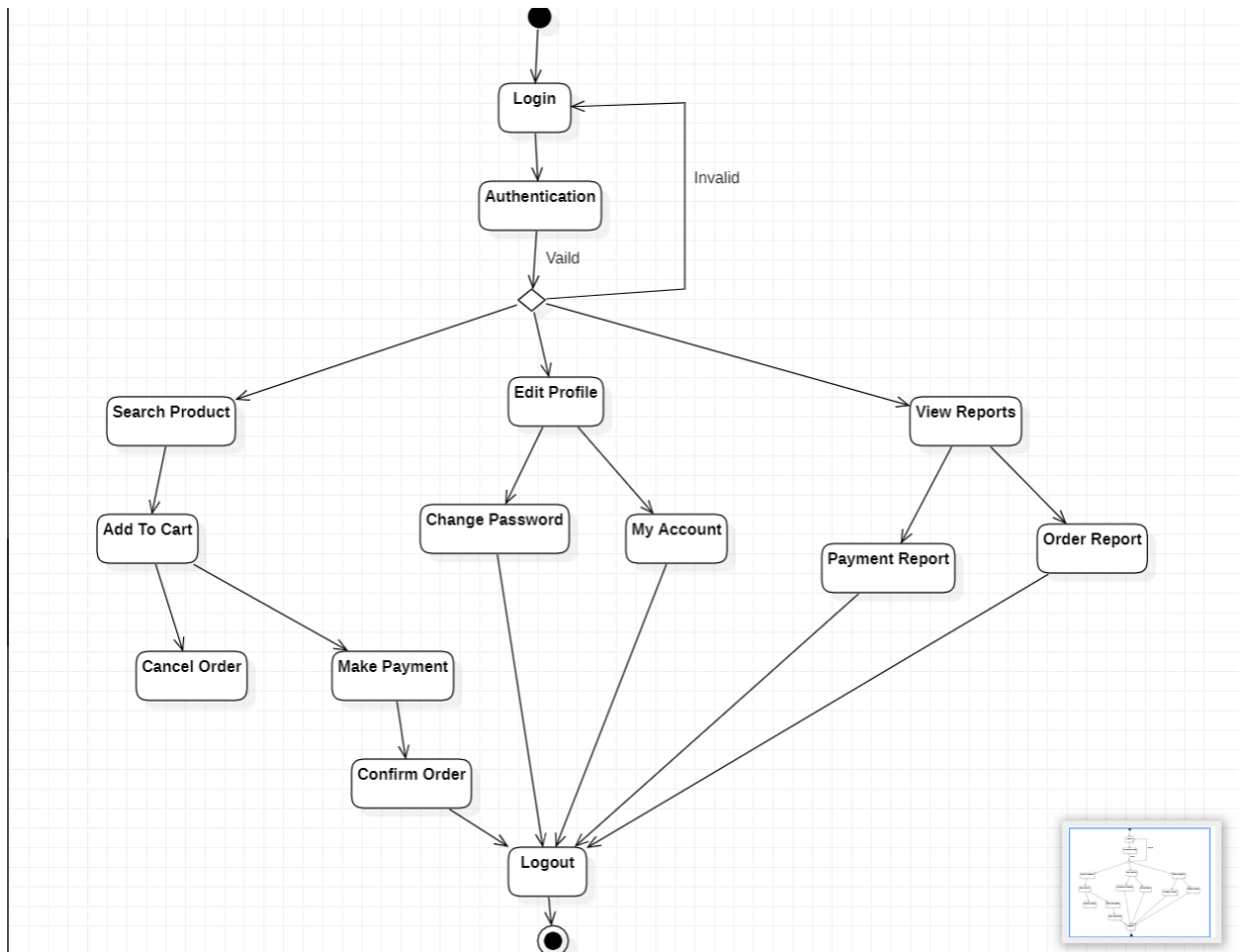
Date: 06/07/2023

Aim : To make state transition diagram for the project

Activity Diagram:

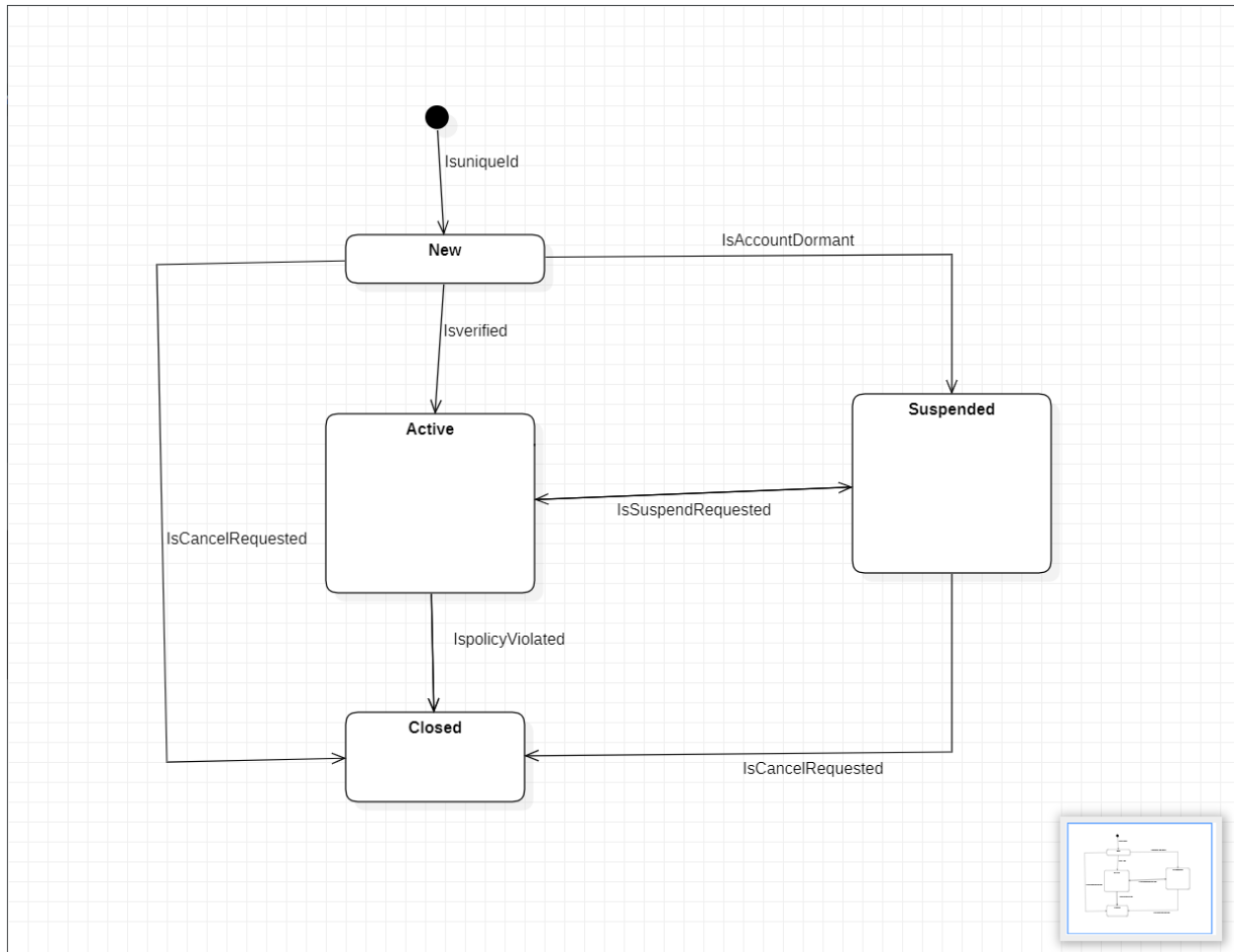
An activity diagram in software engineering is a visual representation of the flow and sequencing of activities or workflows within a system. It illustrates the steps, decisions, and concurrency involved in completing a process. Activity diagrams help analyze and optimize processes, identify bottlenecks, and communicate complex workflows to stakeholders in a clear and concise manner.

Output:



Statechart Diagram: A statechart diagram in software engineering is a graphical representation of the states and transitions that an object or system undergoes over time. It captures the behavior and life cycle of entities, illustrating how they respond to events and change states. Statechart diagrams aid in understanding system behavior, designing control logic, and implementing state-based systems efficiently.

Output:



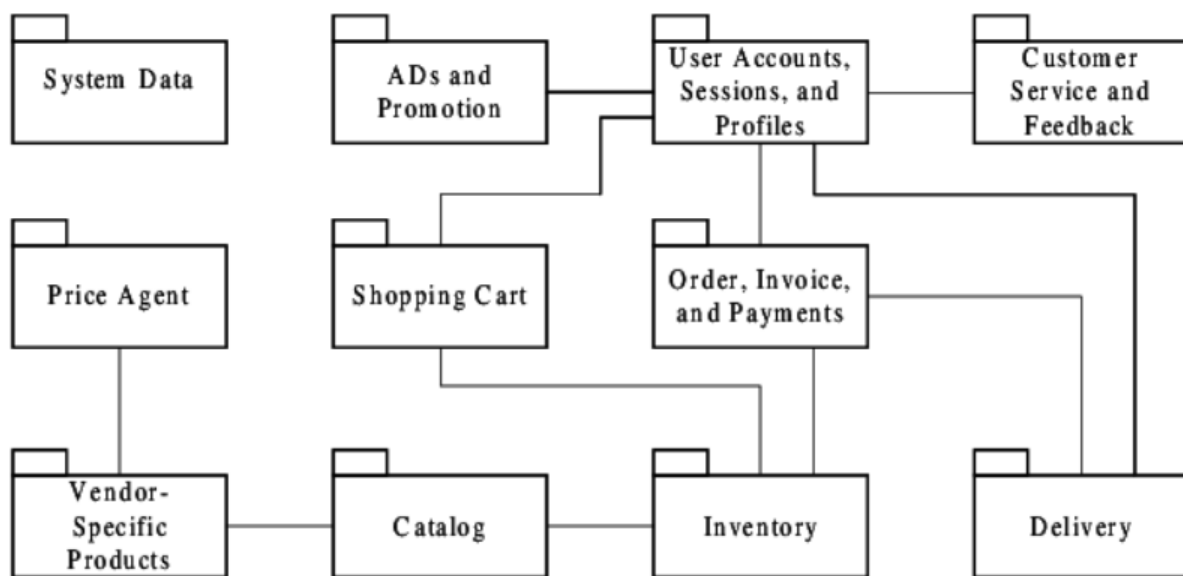
NAME: VAIBHAV SINGH
REG. NO.: 21BCE5616

EXPERIMENT 8

PACKAGE DIAGRAM:

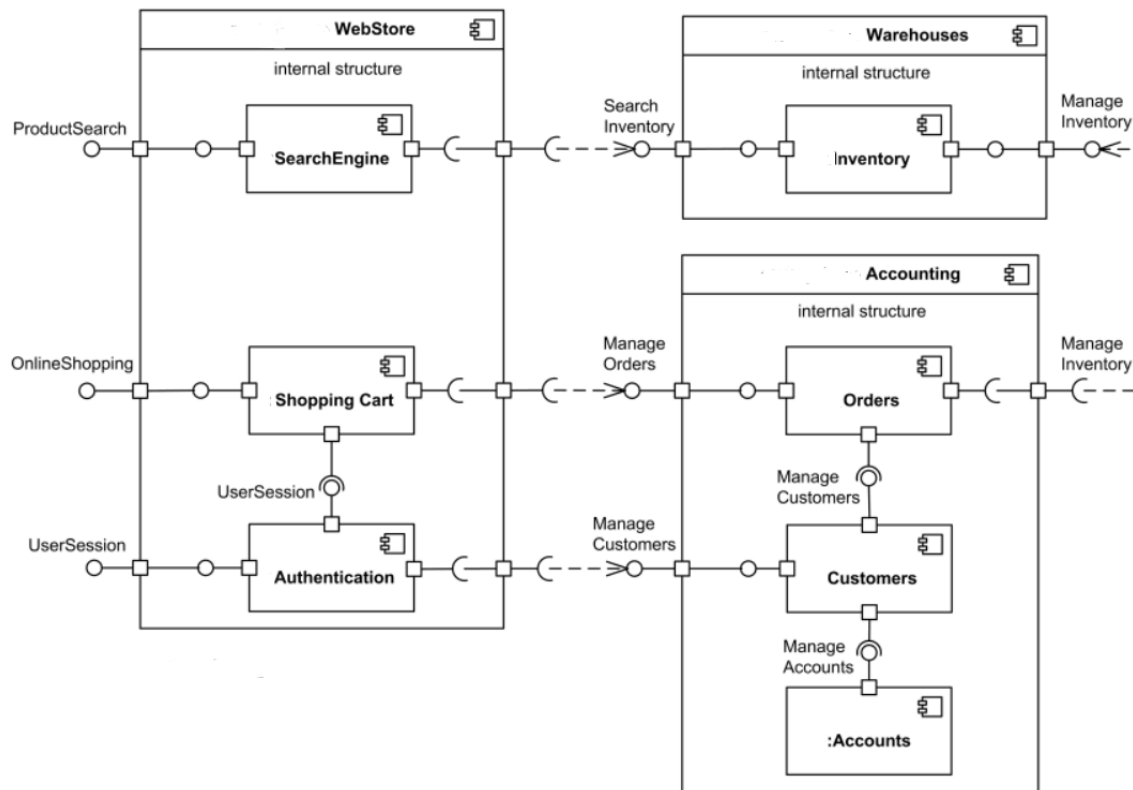
A package diagram is a type of structural diagram in software engineering that represents the organization and dependencies between packages within a software system. It provides a high-level view of the system's modular structure, illustrating how different components are grouped together and interact with each other.

In a package diagram, packages are depicted as rectangles or folders, each representing a logical grouping of related classes, interfaces, or other software artifacts. The dependencies between packages are shown with directed arrows, indicating the flow of dependencies from one package to another.



COMPONENT DIAGRAM:

A component diagram is a type of structural diagram in software engineering that depicts the high-level structure and interrelationships of the components within a system. It provides a visual representation of the software architecture by illustrating the different software components and their dependencies, interfaces, and relationships.



DEPLOYMENT MODEL:

For an e-commerce website, a cloud deployment model is used.

1. **Scalability:** E-commerce websites often experience varying levels of traffic, especially during peak periods such as holidays or sales events. Cloud deployment allows for easy scalability, where additional computing resources can be provisioned on-demand to handle increased traffic.
2. **Cost-Effectiveness:** Cloud deployment eliminates the need for organizations to invest heavily in their own infrastructure. Instead, they can leverage the infrastructure provided by the cloud service provider, paying for the resources they consume on a pay-as-you-go basis.
3. **Reliability and Availability:** Cloud service providers typically offer robust infrastructure with high availability and redundancy. This means that the e-commerce website is less susceptible to single points of failure or downtime.

4. **Global Reach:** Cloud deployment enables organizations to deploy their e-commerce website in multiple regions around the world. This helps reduce latency for users accessing the website from different geographic locations, ensuring a better user experience.
5. **Security and Compliance:** Cloud service providers offer robust security measures and compliance certifications, which can be beneficial for e-commerce websites that handle sensitive customer data.