Retail Product Management System

	Prepared By / Last Updated By	Reviewed By	Approved By
Name	1.Manish Shingare 2.Priyanka Jatav 3.Dhriti Sinha 4.Ashi Agrawal 5.Prajwal Nirmal 6. Himanshu Ranjan Singh		
Role	Manish Shingare (2125636) – POD Lead Priyanka Jatav (2125801) – POD Member Dhriti Sinha (2125946) – POD Member Ashi Agrawal (2125257) – POD Member Prajwal Nirmal (2125331) – POD Member Himanshu Ranjan Singh (2125459) -- POD Member		
Signature			
Date			

Table of Content

1.0 Introduction **(3)**

1.1 Purpose of This Document
1.2 Project Overview

2.0 Solution Summary **(4)**

2.1 Scope
2.2 Assumption
2.3 Risk
2.4 Hardware and Software Requirements

3.0 System Designing **(6)**

3.1 Schematic Diagram
3.2 Flow Chart
3.3 System Architecture Diagram

4.0 Functional Requirements and High Level Designing **(8)**

4.1 Individual Components of the System
4.1.1 Product Microservice
4.1.2 ProceedToBuy Microservice
4.1.3 Vendor Microservice
4.1.4 Authorization Microservice
4.1.5 Swagger
4.1.6 E-Commerce Portal (MVC)

5.0 Data Base Designing **(13)**

6.0 Cloud Deployment requirements **(13)**

1.0 Introduction

1.1 Purpose of this document

The purpose of the software requirement document is to systematically capture requirements for the project and the system. "Retail Product Management System" that has to be developed. Both functional and non-functional requirements are captured in this document. It also serves as the input for the project scoping.
The scope of this document is limited to addressing the requirements from a user, quality, and non-functional perspective.
High Level Design considerations are also specified wherever applicable, however the detailed design considerations have to be strictly adhered to during implementation.

1.2 Project overview

A leading Retail Organization wants to strengthen its Middleware by exposing the core logic related to Product Management as Microservices. This middle ware Microservices will be hosted on Cloud so that all the up/downstream applications can get access to this for performing business transactions.
There will also be an e-Commerce Portal to be developed part of this scope that consumes these Microservices and responses back to customers who are purchasing a product

2.0 Solution Summary

2.1 Scope

Below are the modules that need to be developed as part of the Project:

Req. No.	Req. Name	Req. Description
REQ_01	Products Module	Products Module is a Middleware Microservice that performs the following operations: Search Product by Id Search Product by Name Add Rating to Product
REQ_02	ProceedToBuy Module	ProceedToBuy Module is a Middleware Microservice that performs the following operations: Customer can add Product to wish list – this is when product is out of stock Add Product to cart
REQ_03	Vendor Module	Vendor Module is a Middleware Microservice that performs the following operations: Give List of Vendors for the given Product
REQ_04	Ecommerce Portal	An e-Commerce Portal that allows a customer to Login and allows to do following operations: Login Search for a product by id or name. In the Product page, key in zip code and expected delivery date Proceed To Buy. If product is out of stock, add it to wish list Display product details (Id, Name, Price, Vendor details, Delivery charge, Order Total) in checkout page. Rate a Product

2.2 Assumptions

- This project is a stand-alone project so it will not affect the system where it will be embedded.
- This system will not depend on any other module. It will be a web-based so everyone will independently contact it.
- It is will not affect the environment at all.

2.3 Risks

- Operational Risk: Operational risk arises from the potential for loss due to significant deficiencies in system reliability or integrity.
- Legal Risk: Legal risks can arise in relation to the access to, use and dissemination of data and information.
- Reputational Risk: Reputational risk means the risk for the library to experience significant negative public opinion that may result in "losing popularity" with existing and potential customer.

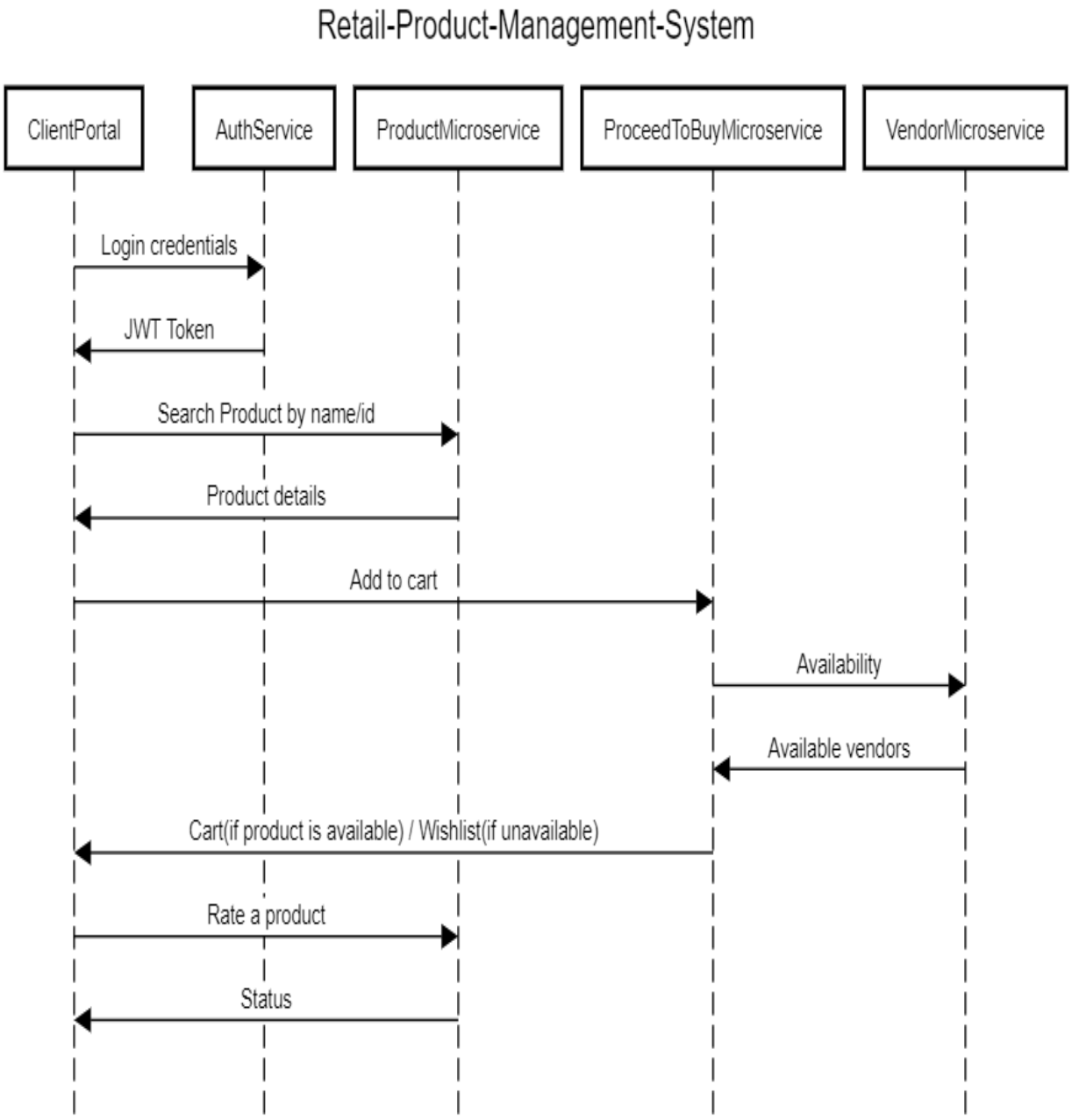
2.4 Hardware and Software Requirement

- Hardware Requirement:
- Developer Desktop PC with 8GB RAM
- Software Requirement (Asp.Net Core)
- Docker (Optional)
- Postman Client in Chrome
- Software Requirement (Dotnet)
- Visual studio 2019 enterprise edition
- SQL Server 2015
- Azure cloud access

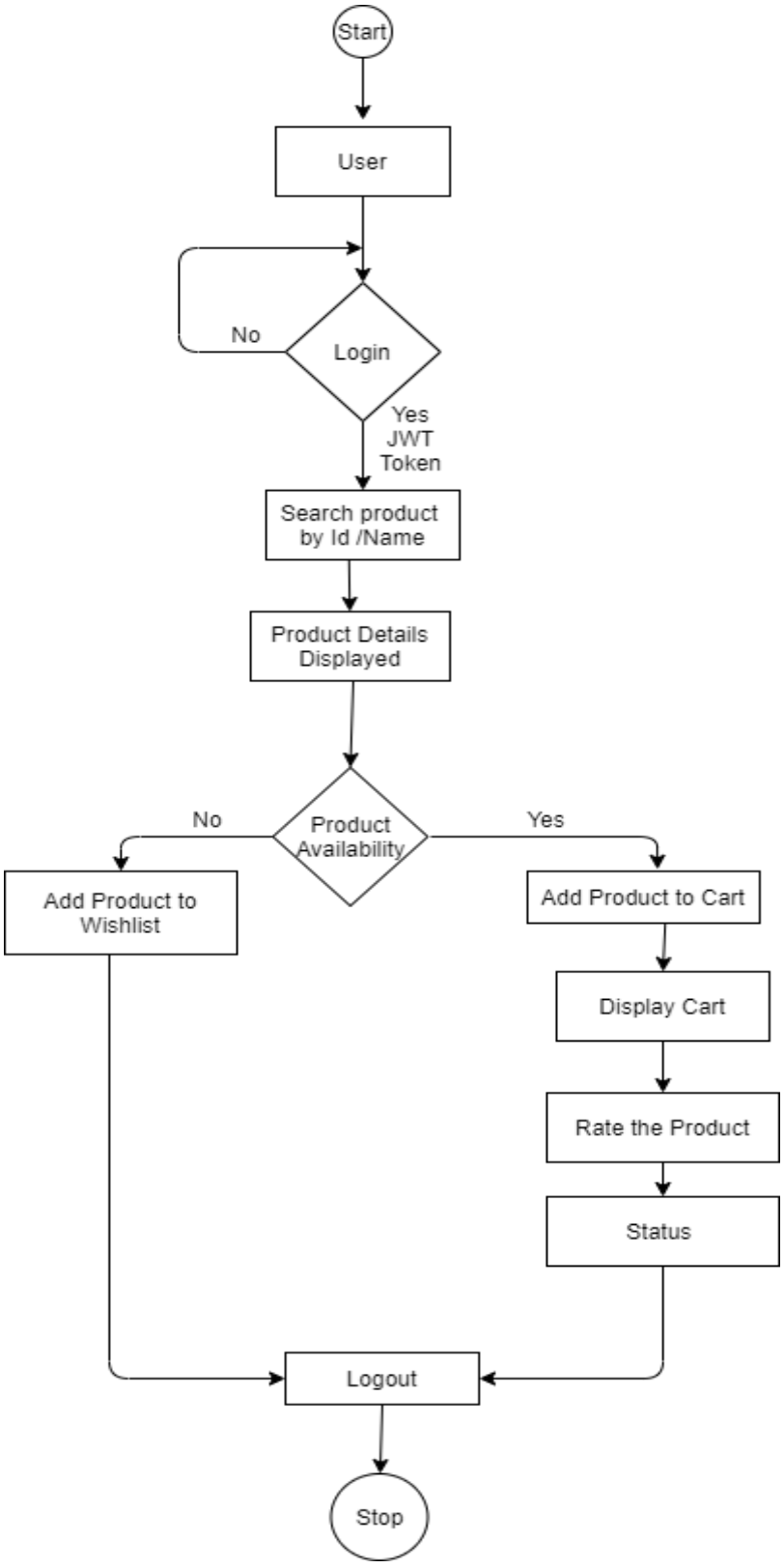
3.0 System Designing

3.1 Schematic Diagram

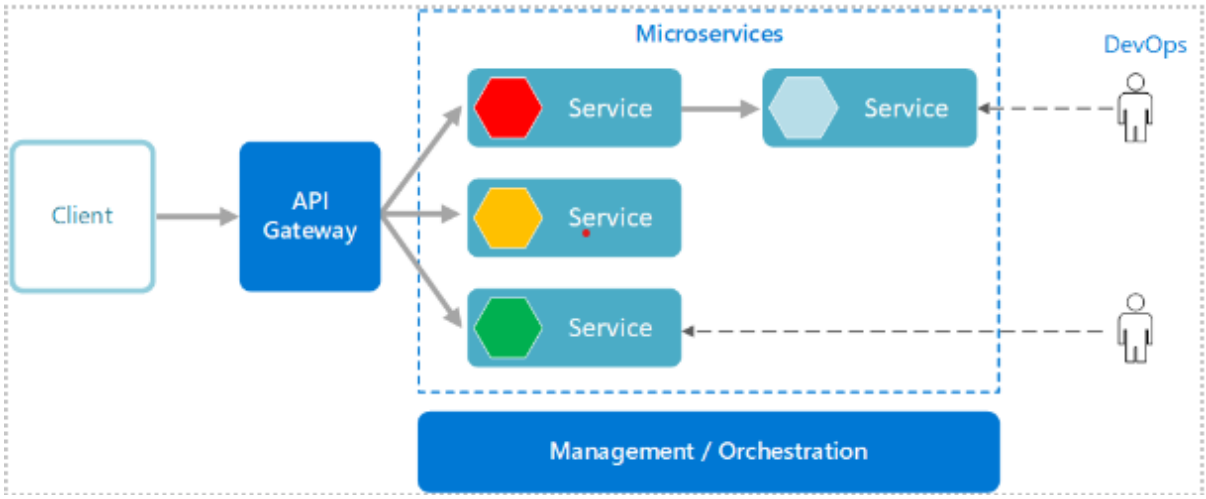
A schematic, or schematic diagram, is a representation of the elements of a system using abstract, graphic symbols rather than realistic pictures. It gives an overview of the overall system



3.2 Flow Chart



3.3 System Architecture Diagram



4.0 Functional Requirements and High Level Design

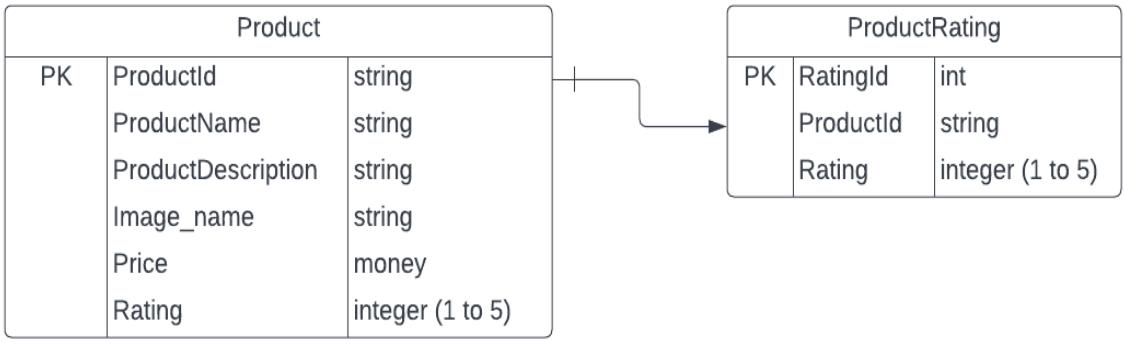
4.1 Individual Components of the System

4.1.1 Product Microservice

Product Management System	Product Microservice
Functional Requirements You can assume that eCommerce Portal App is the only client to this Microservice. An authorized customer can search for a product using a product id or name. Post Authorization, the product id, name, price, product image reference and description details are returned as output. If the product is inactive or out of stock then appropriate message should returned as output. The customer adds a Rating which must get associated to the product	
Entities ○ **Product** <Id, Price, Name, Description, image name, rating> **REST End Points** **Product Microservice**	

○ GET: /searchProductById (Input: Product_ID | Output: Product object) ○ GET: /searchProductByName (Input: Product_Name| Output: Product object) ○ POST: /addProductRating (product ID, rating | Output: Status)
Trigger – Can be invoked from e-Commerce Portal (local MVC app)
Steps and Actions 1. Customer Portal will request for searchProductById, searchProductByName operations. 2. Authorization has to be performed. 3. At any point in time, the customer must be able to search for a product detail from the eCommerce Portal Client. Hence the Product Microservice must expose the Product Details through /searchProductByID and /searchProductByName REST End Point. 4. Customer must be able to add Rating to a product which must get associated to the Product entity, for the respective product
Non-Functional Requirement: • Only Authorized Customer can access these REST End Points

ER-Diagram

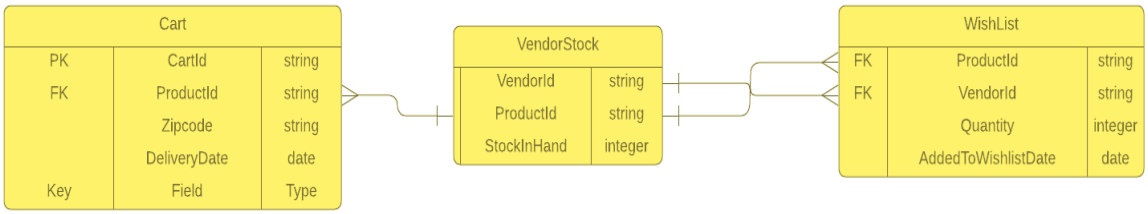


4.1.2 ProceedToBuy Microservice

Product Management System	ProceedToBuy Microservice
Functional Requirements Can assume that eCommerce Portal App is the only client to this Microservice. An authorized customer can add the product to the cart. An authorized customer can add product to cart and view the product details (Product Price, Delivery Date, Vendor details) in the checkout/cart page. If Product is not available, the customer can add the product to Wishlist. Post Authorization, ProceedToBuy Microservice will interact with the Vendor Microservice for the following functionalities:	

The Microservice will interact with Vendor Module, to get a assign a Vendor for a given order.
Entities ○ **Vendor** <Vendor ID, Vendor Name, Delivery Charge> ○ **Cart** <CartId, ProductId, zipcode, Delivery Date, Vendor object> ○ **Vendor Wishlist** <Vendor ID, Product ID, Quanitity, Date Added to Wishlist etc.> **REST End Points** **Cart Microservice** ○ POST: /addProductToCart (Input: Customer_ID, Product_ID, Zip_Code, Expected_Delivery_Date | Output: Cart) ○ POST: /addProductToWishlist (Input: Customer_ID, Product_ID | Output: Status)
Trigger – Can be invoked from e-Commerce Portal (local MVC app)
Steps and Actions 1. Customer Portal will request for ProceedToBuy operations. 2. For all the operations, the Customer Profile will be verified before interacting with other Microservices. 3. In product page customer should key in zip code and select expected delivery date. On click of submit /addProductToCart end point is invoked, the product is added to cart and the user is directed to the cart page. The cart details are displayed to the user there. 4. Internally, in the /addProductToCart service method we will invoke the /getVendorDetails end point of the Vendor microservice to get the appropriate vendor to fulfill the order. The response returned by Vendor has to be cascaded as the response in this end point. 5. Appropriate vendor must be tagged to the Product based on vendor rating.
Non-Functional Requirement: • Only Authorized Customer can access these REST End Points.

ER-Diagram



4.1.3 Vendor Microservice

Product Management System	Vendor Microservice
Functional Requirements During the order fulfillment cycle, a vendor will be mapped to the order. The vendor will ship the product from their warehouse and deliver it to the customer location.	

ProceedToBuy Microservice interacts with Vendor Microservice to assign a vendor for the customer's order. Post authorization of request, Vendor Microservice allows the following operations:

Choose the Vendor for the given product, who has enough quantities to supply
Perform the below mentioned checks to select a vendor

Vendors should have the given product in stock. If available, retrieve all vendors along with the rating and return to the ProceedToBuy service

Assume that the vendor details will be stored in system automatically, which is out of scope of the system.

Entities

- **Vendor**

<Vendor ID, Vendor Name, Delivery Charge, Rating>

- **Vendor Stock**

<Product ID, Vendor ID, Stock in Hand, Expected Stock Replenishment Date, etc.>

REST End Points
Vendor Microservice

- GET: /getVendorDetails (Input: Product_ID | Output: Vendor ID, Vendor Name, rating, Delivery Charge)

Trigger – Can be invoked from ProceedToBuy Microservice

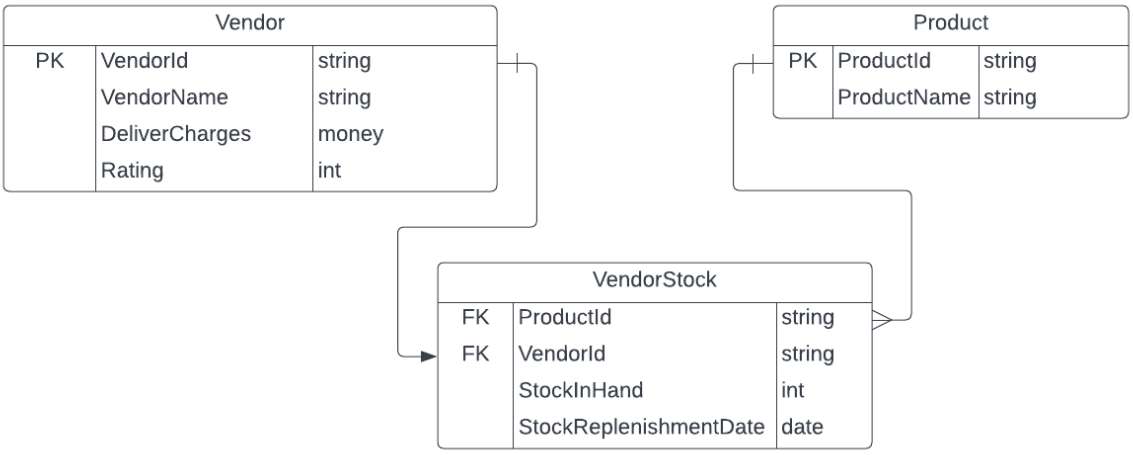
Steps and Actions

1. Vendor Microservice will have 1 End Points exposed to ProceedToBuy Microservice
2. If /getVendorDetails end point is invoked by ProccedToBuy Microservice, the Vendor Microservice will perform the necessary checks w.r.t stock and will return the response back to ProccedToBuy Microservice.

Non-Functional Requirement:

- Only Authorized Customer can access these REST End Points.
- If there is more than 1 customer adding the product to cart and proceeding to buy, the tasks have to be handled in parallel

ER-Diagram



4.1.4 Authorization Microservice

Product Management System	Authorization Microservice

Security Requirements

- Service to Service communication has to happen using JWT
- Pass End User Context across Microservices
- Have the token expired after specific amount of time say 15 minutes.
- Have this service configured in the cloud along with other services

4.1.5 Swagger

Product Management System	Swagger

Documentation Requirements

- All the Microservices must be configured with Swagger for documentation

.Net implementation

- Register the swagger resources in the Swagger Microservice and enable them as REST end points
- Configure this service along with other services in the cloud

4.1.6 e-Commerce Portal (MVC)

Product Management System	e-Commerce Portal

e-Commerce Portal Requirements

An e-Commerce Portal that allows a customer to Login and allows to do following operations:

Login

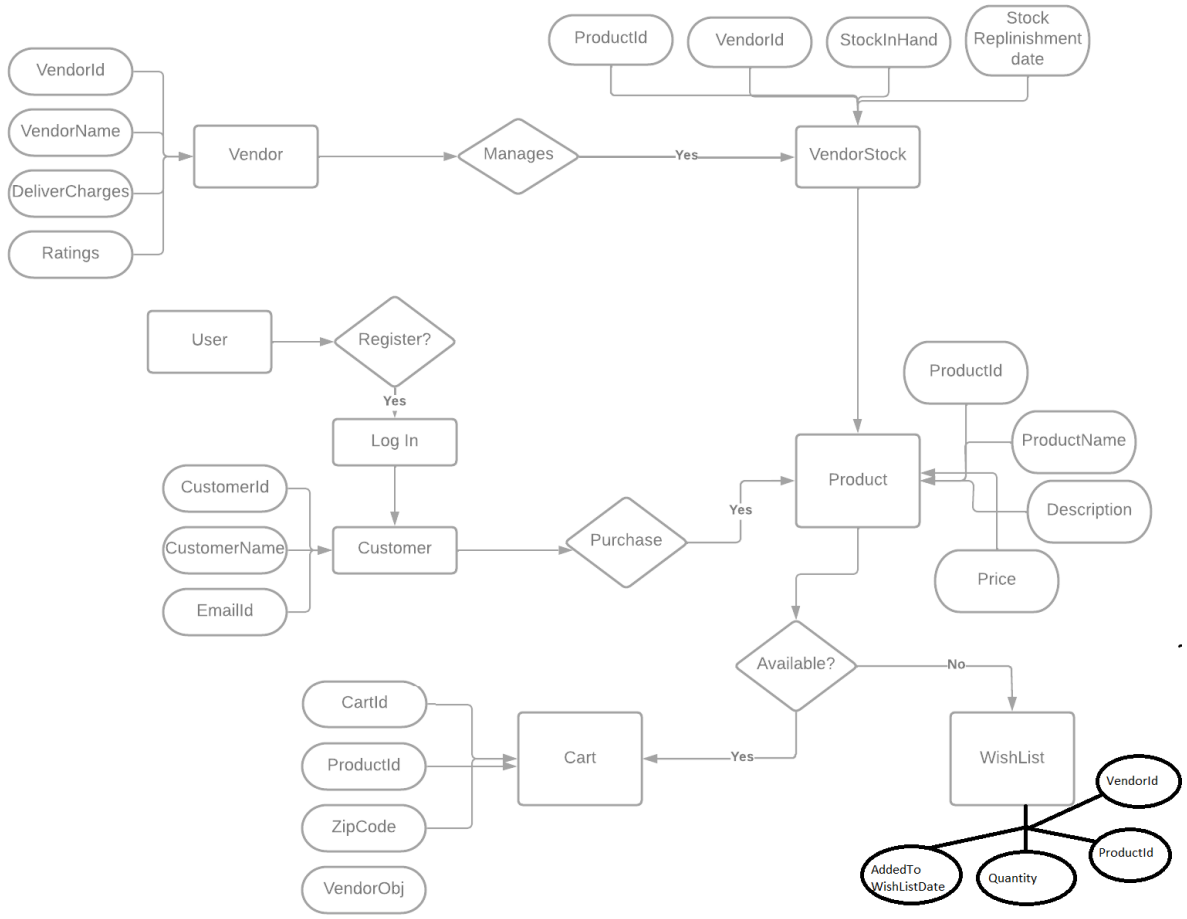
Search for a product by id or name.

In the Product page, key in zip code and expected delivery date.

Proceed To Buy.
Add Product to Wishlist
- Display product details (Id, Name, Description, Price, Vendor details,Delivery charge, Order total) in checkout page.

Each of the above operations will reach out to the middleware Microservices that are hosted in cloud.

5.0 Data Base Designing



6.0 Cloud Deployment requirements

- All the Microservices are deployed in the Cloud
- All the Microservices are independently deployable. They have to use In-memory database or user sessions wherever applicable
- The Microservices are dockized and these containers must be hosted in Cloud using CI/CD pipelines
- The containers are orchestrated using Azure Kubernetes Services.
- These services are consumed from an MVC app running in a local environment.