Software Testing Assignment

Module- 1(Fundamental)

Q-1. what is SDLC

ANS.. The software development life cycle(SDLC) is the cost-effective and time-efficient process that development teams use to design and build high-quality software. the goal of SDLC is to minimize project risks through forword planning so that software meets customer expectations during production and beyond. this methodology outlines aseries of steps that divide the software development process into tasksyou can assign, complate, and measure.

- SDLC Phases: Planning

  Analysis

  Design

  Build

  Testing

  Deploy

  Maintain

Q-2 What is software testing?

ANS.. Software testing is the process of finding errors in the developed product. it also checks whether the real outcomes can match expected results, as well as aids in the identification of defects, missing requirements, or gaps.

testing is the penultimate step befor the launch of the product to the market. it includes examination, analysis, observation, and evaluation of different aspects of product.

professional software testers use a combination of manual testing with automated tools. after conducting tests, the testers report the

results to the development team. the end goal is to deliver a quality product to the customer, which is way software testing is so important.

Q-3 **What is agil methodology?**

ANS.. Focus on process adaptability and customer satisfaction by rapid delivery of working software product.

break the product into small incremental builds.

these builds are provided in iteration every iteration involves cross functional team working simultaneouslu on various areas like planning, requirements analysis, design, coding, unit testing, and acceptance testing.

at the end of the iteration aworking product is displayed to the customer and important stakeholders

agile methodology is a project management framework that breaks projects down into several dynamic phases, commonly known as sprints.

the agile framework is an iterative methodology. after every sprint, teams reflect and look back to see if there was anyting that could be improved so they can adjust their strategy for the next sprint.

Agile methodology:

1. plane

2. design

3. develop

4. test

5. deploy

6. review

Q-4 What is SRS

ANS.. A software requirements specification (SRS) is a document that describes what the software will do and how it will be expected to perform.it also describes the functionality the product needs to fulfill the need of all stakeholders(business, users).

you can think of an SRS as a blueprint or roadmap for the software you're going tobuild. the elements that comprise an SRS can be simply summarized into four ds:

- Define your product's purpose.
- Describe what you're building.
- Detail the requirements.
- Deliver it for approval.

We wamt to DEFINE the purpose of our product, DESCRIBE what we are building, DETAIL the individual requirements,and DELIVER it for approval. a good SRS document will define everything from how foftware will interact when embedded in hardware to the expectation when connected to other software .an even better SRS document also accounts for the needs of real-life users and human interaction.

Q-5 What is OOPS

ANS.. Object oriented programing is a programming paradigm based on the concept of "objects".

an object can be defined as a data field that has unique attributes and behaviour.

OOP is an object-oriented programming technique that combines date and instructions for processing that data into an object-oriented programming provides concepts that help modelling complicated system of real world into manageable software solutions.

Object -oriented programming as the name suggests, object-oriented programming or OOPs refers to languages that use object in programming,they use objects as a primary source to implement what is to happen in the code. object are seen by the viewer or user, performing tasks assigned by you.

Q-6 Write basic concepts of OOPs

ANS.. Object oriented programing is a programming paradigm based on the concept of "objects".

an object can be defined as a data field that has unique attributes and behaviour.

concept of OOP:

OBJECT

CLASS

ENCAPSULATION

INHERITANCE

POLYMORPHISM

ABSTRACTION

- OBJECT: a think that can be seen and touched. but is not alive.

anything that is visible or tangible and is relatively stable in form. a thing, person, or matter to which thought or action is directed: an object of medical investigation.

- CLASS: Object oriented programming (OOP) is a programming paradigm based on concepts of object. the objects the may contain data as a form of instance variables and behaviours in form of method.

- ENCAPSULATION: Encapsulation is basically information hiding it describes the idea of bundling data and methoods that work on that data within one unit. in here access to data need to be controlled using access modifiers (public, private, protected etc..) and expose them to the outside world using getters and setters.

- INHERITANCE: Inheritance means sub-class inherits from the super-class. in animal class, there are methods and attributes that are common to all animals by using inheritance concept other child classes can use those attributes and methods on the parent class.

- POLYMORPHISM: Polymorphism is a feature of object-oriented programming languages that allows a specific routine to use variables of different types at different times.
- ABSTRACTION: Abstract classes and methods. it is not allowed to create an instance of a class that has been defined as abstract. any class that contains at least one abstract method.

Q-7 What is object

ANS.. Object oriented programming (OOP) is a programming paradigm based on concepts of object. the objects the may contain data as a form of instance variables and behaviours in form of method. anything that is visible or tangible and is relatively stable in form.

a thing, person, or matter to which thought or action is directed.

Q-8 What is class

ANS.. Object oriented programming (OOP) is a programming paradigm based on concepts of object. the objects the may contain data as a form of instance variables and behaviours in form of method. A class describes the contents of the objects that belong to it: it describes an aggregate of data fields, and defines the operation.

Q-9 **What is encapsulation**

ANS.. Encapsulation is basically information hiding it describes the idea of bundling data and methods that work on that data within one unit. in here access to data need to be controlled using access modifiers (public, private, protected etc..) and expose them to the outside world using getters and setters. A way to restrict the direct access to some components of an object, so users cannot access state values for all of the variables of a particular object.

Q-10 **What is polymorphism**

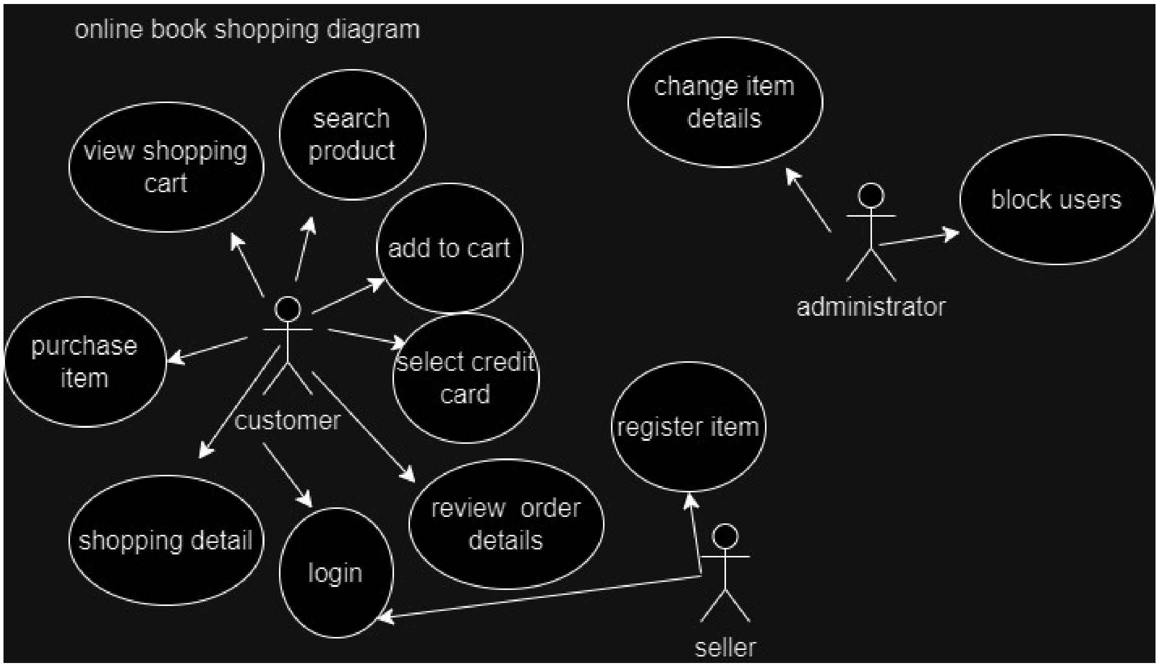
ANS.. Polymorphism is a feature of object-oriented programming languages that allows a specific routine to use variables of different types at different times. Polimorphism is a feature of object-oriented programming languages that allows a specific routine to use variables of different types at different times. polymorphism in programming in programming gives a program the ability to redefine methodes for derived classes.

Q-11 **What is inheritance**

ANS.. Inheritance means sub-class inherits from the super-class. in animal class, there are methods and attributes that are common to all animals by using inheritance concept other child classes can use those attributes and methods on the parent class. Inheritance refers to the assets that an individual bequeaths to their loved ones after they pass away. an inheritance may contain cash, investments such as stocks or bonds, and other assets such as jewelry, automobiles, art, antiques, and real estate.

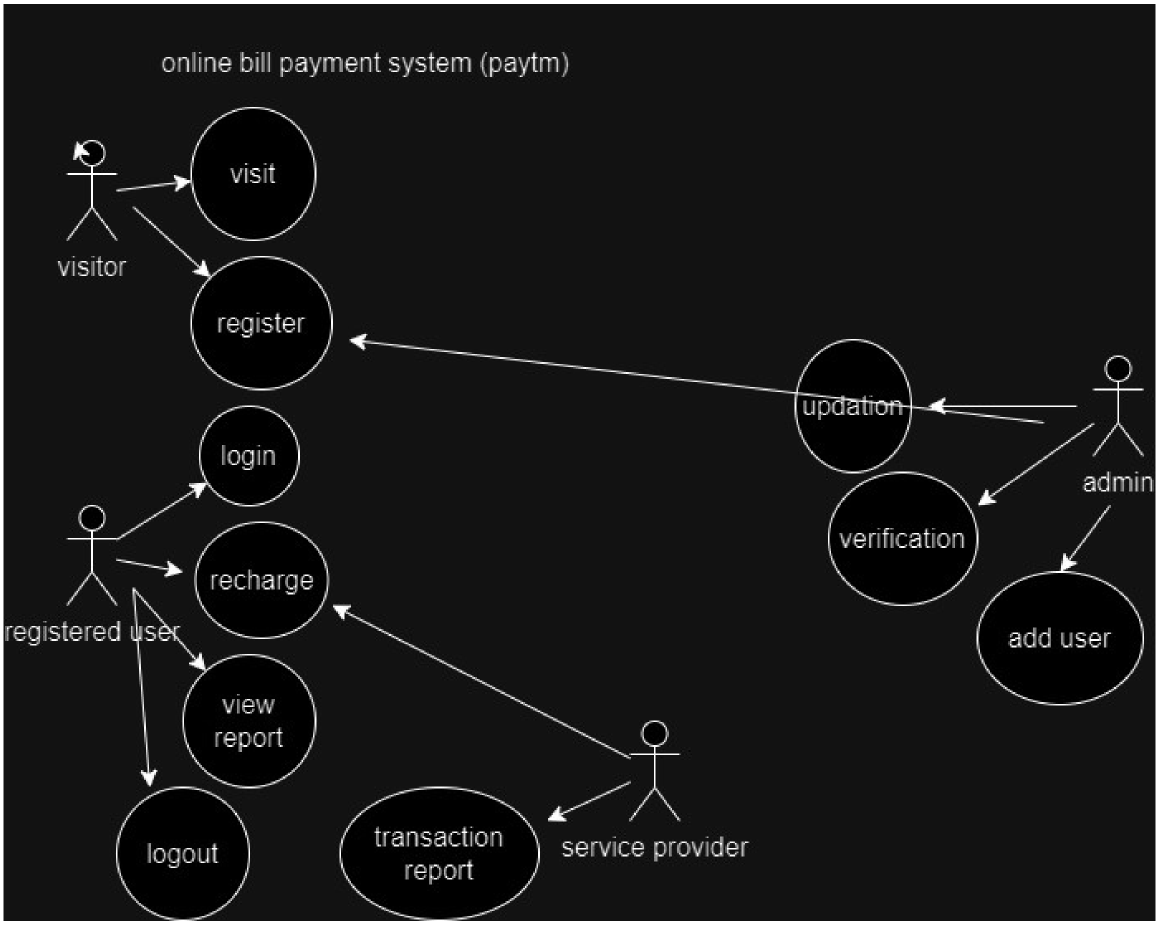
Q-12 Draw usecase on online book shopping

ANS..



Q-13 Draw usecase on online bill payment system (paytm)

ANS..



Q-14 Write SDLC phases with basic interoduction

ANS.. SDLC phases:

planning (requirement gethering)

analysis

design

build

testing

deploy

maintain

- REQUIREMENT GATHERING: the software development team work to carry on the project

The team holds discussions with various staceholders form problem domain and tries to bring out as much information as possible on their requirement.

types of requirements: Function requirements

describe system services or functions.

Non-functional requirement : describe constraints on the system

or the development process.

- ANALYSIS PHASE: It defines the requirements of the system indeoendents of how these requirement will be accomplished

it is carried out by the senior developers / testers of the team with information from the client, the pre-sales, market studies and domain specialists of the industry

- DESIGN PHASE: In this third phase , software design documents are prepared as per the requirement specification document. this help define overall system architecture.

2 kinds of design documents developed in this phase.

1. high-level design(HLD)

2. Low level design(LLD)

- IMPLEMENTATION /CODING PHASE: This stage has many names such as the build, development, coding or implemention phase developers start build the entire system by writing code using the chosen programing language.

tasks are divided into units or modules and assigned to the various developers.

longest phase of the software development life cycle process.

- TESTING PHASE: Qa (quality assurance) validation phase testing is defined as an activity to check whether the actual results match the expected results and to ensure that the software system is defect free.

the testing team starts testing the functionality of the entire system.

qa and testing team may find some bugs/defects which they communicate to developers.

the development team fixes the bug and send back to qa for a re teas

- DEPLOYMENT PHASE: Acceptance or beta evalution phase it includes a release specifically for a market facing group of people and gets.it tested in a real-time environment for their aceptance. it is a soft of user acceptance testing

focuses on fixing some usability bugs or enhancement crucial for the market perspective or can also give a green status for delivering it to the target customers.

- MAINTENANCE PHASE: After the software clears all the sdlc phases without any issues, then it goes into the maintenance stage.

it allows the customers to request for upgrades and get the fixes/patches for problem internally or externally identified.

maintenance is the proccess of changing a system after it has been deployed.

corrective maintenance:- identifing and repairing defects.

Adaptive maintenance:- adapting the existing solution to the new platforms.

perfective maintenance:- implementing the new requirements.

Q-15 Explain phases of the waterfall model

ANS.. Requirements/analysis

design

coding

testing

maintenance

It is simple but idealistic

other software development life cycle models are based on the classical waterfall model.

in this model one phase can be started after completion of the previous phase.

the outout of one phase will be the input to the next phase.

development process can be considered as a sequential flow in the waterfall.

requirements must be "frozen" to early in the life cycle.

requirement s are validated too late .

requirement s are very well documented, clear and fixed,no ambiguous requirements that have to be followed.

in below applications, the requirements are know well in advance and contracts are very specific about the deliverable of the project.

real use of waterfall model: development of department of defense military and aircraft programs followed waterfall model in many organizations

"calculator"

Q-16 <u>Write phases of spiral model</u>

ANS.. Combination of iterative development process model and waterfall model with a very high emphasis on risk analisis.

spiral model phases :-

- PLANNING: it includes estimating the cost, schedule and resources for the iteration it also involves understanding the system requirements for continuous communication between the system anakyst and the customer.
- RISK ANALYSIS: identification of potential risk is done while risk mitigation strategy is planned and finalized.
- ENGINEERING: it includes testing, coding and deploying

software at the customer site.

- EVALUTION: evalution of software by the customer.also includes identifying and monitoring risk such as schedule slippage and cost overrun.

When to use spiral model :

when project is large.

when releases are required to be frequent

when creation of a prototype is applicable

whan risk and costs evalution is important

for medium to high-risk projects.

when requirements are unclear and complex

when changes may require at any time

when long team project commitment is not feasible due to changes in economic priorities

Q-17 Write agile manifesto principles..

ANS.. Individuals and interactions-

more important is given to agile team members and individuals associated with the product rather than focusing and concentrating more upon traditional processes and tools, or technology used in the production process.

development of working software takes precedence over detailed decumentation and paperwork demo working software is considered the best means of communication with the customer to understand their requirement, instead of just depending on documentation.

customer collaboration: as the requirements cannot be gathered completely in the beginning of the project due to various fsctors, continuous customer interaction is very important to get proper product requirement

risponding to change: focused on quick responses to change and continuous development.

Q-18 Explain working methodology of agile model and also write pros and cons..

ANS.. Focus on process adaptability and customer satisfaction by rapid delivery of working software product.

break the product into small incremental builds.

these builds are provided in iteration every iteration involves cross functional team working simultaneouslu on various areas like planning, requirements analysis, design, coding, unit testing, and acceptance testing.

at the end of the iteration aworking product is displayed to the customer and important stakeholders

agile methodology is a project management framework that breaks projects down into several dynamic phases, commonly known as sprints.

the agile framework is an iterative methodology. after every sprint, teams reflect and look back to see if there was anyting that could be improved so they can adjust their strategy for the next sprint.

Agile working is about bringing people, processes, connectivity and technology, time and place together to find the most appropriate and effective way of working to carry out a particular task. it is working within guidelines but without boundaries.

PROS:

Functionalily can be developed repidly and demonstrated.

suitable for fixed or changing requirements delivers early partial working solutions. minimal rules, documentation easily employed littele or no planning required error can be fixed in the middle or the project.

CONS:

Not suitable for handling complex dependency an overall plan , an agile leader and agile practice is a must without which it will not work

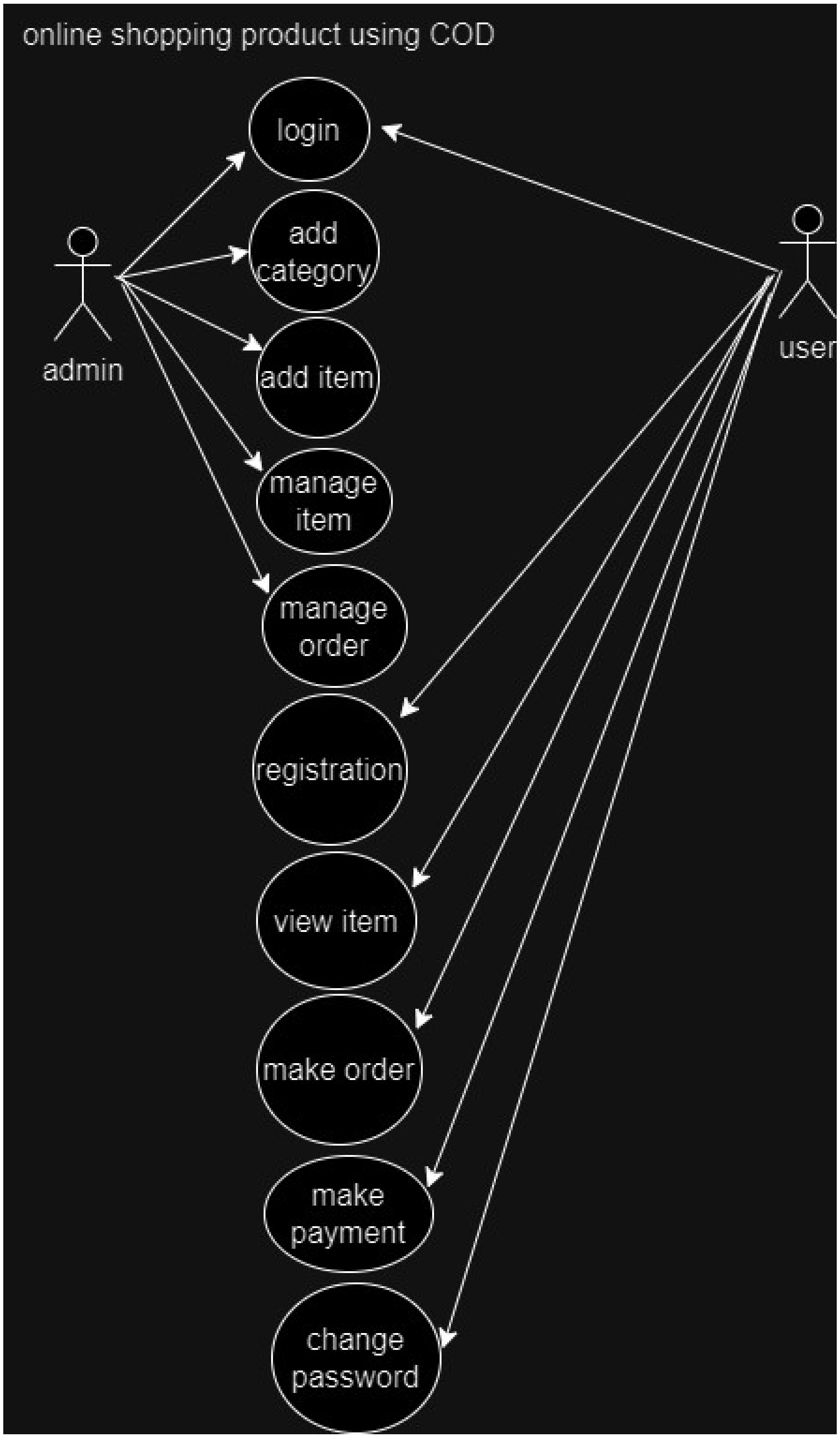
depends heavily on customer interaction so of customer is not clear, team can be driven in the wrong direction.

there is very high individual dependency, since there is mimimum documentation generates.

transfer of technology to new team members may be quite cgallenging due to lake of documentation.

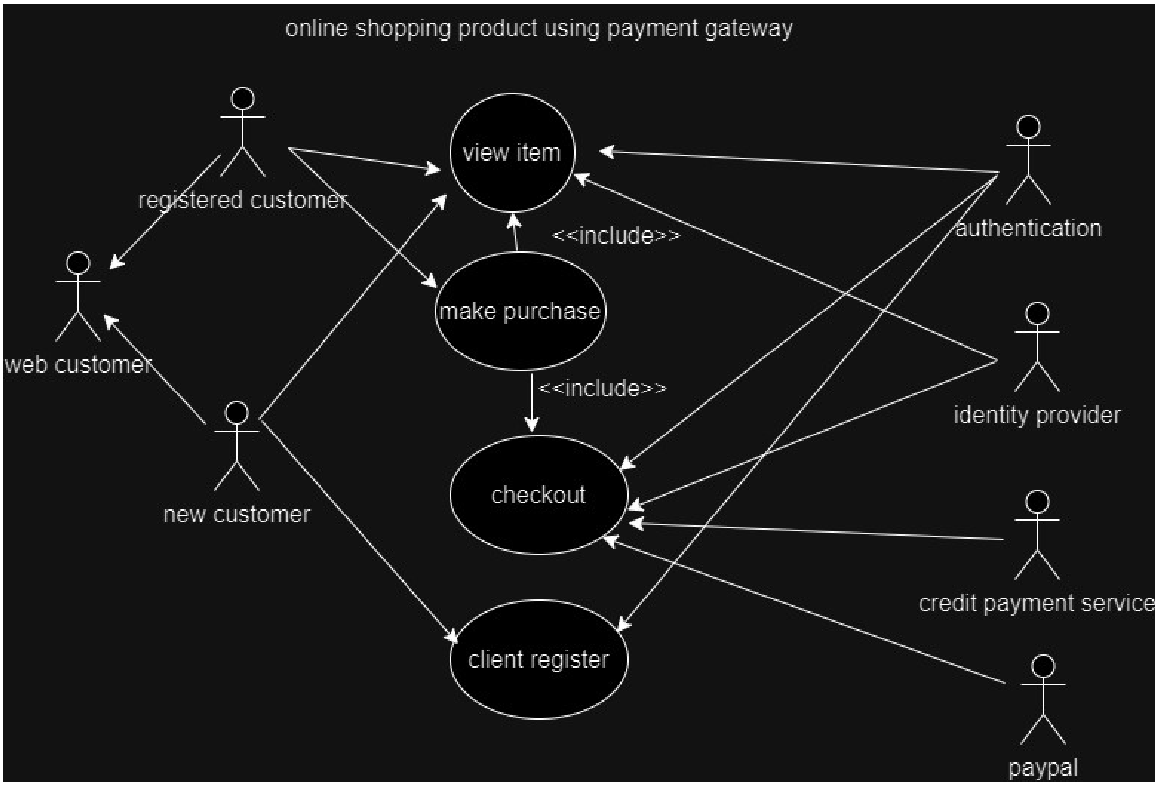
Q-19 Draw usecase on online shopping product using COD.

ANS..


online shopping product using COD
admin
user
login
add category
add item
manage item
manage order
registration
view item
make order
make payment
change password

Q-20 Draw usecase on online shopping product using payment gateway.

ANS..


online shopping product using payment gateway
registered customer
web customer
new customer
view item
<<include>>
make purchase
<<include>>
checkout
client register
authentication
identity provider
credit payment service
paypal