Module -1 (fundamental)

Q1. What is SDLC ?

Software development lifecycle

SDLC is a structure imposed on the development of a software product That defines the process for planning, implementation, testing, Documentation, deployment, and ongoing maintenance and support. There are a number of different development models.

Q-2. what is software testing

Testing is a process used to identify correctness, completeness and quality of developed computer software

Q3. What is agile methodology?

Iterative approach is taken and working software build is delivered after each iteration. Each build is incremental in terms of features; the final build holds all the features required by the customer

Q4. Write SDLC phases with basic introduction

(1) Requirement gathering (2) Analysis (3) Design (4) Implementation (5) Testing (6) Maintenance

1.Requirement gathering

. Build constant feedback into the project plan

. Plan for change

. Early prototyping [e.g., UI] can help clarify the requirements

. Functional and Non-Functiona

2. Analysis

This analysis represents the “what” phase.

The deliverable result at the end of this phase is arequirement document.

This phase represents the “how” phase

The deliverable design document is the architecture.

3. Design

. Design Architecture Document

. Implementation Plan

. Critical Priority Analysis

. Performance Analys

4. Implementation

In the implementation phase, the team builds the components either From scratch or by composition.

The implementation phase deals with issues of quality, performance, baselines, libraries, and debugging.

The end deliverable is the product itself. There are already many established techniques associated with implementation.

5. Testing

Tester, Market production

It is much easier to explain to a customer why there is a missing feature than to explain to a customer why the product lacks quality.

A customer satisfied with the quality of a product will remain loyal and wait for new functionality in the next version.

6. Maintenance

1)Corrective: finding the defect and correct it

2)Adaptive: take it to new platform

Adapting the existing solution to the new Platforms

3)Perfective: make it more perfect implementing the new requirements In a spiral lifecycle, everything after the delivery and deployment of the first prototype can be considered "maintenance"!

Q-5 Explain Phases of the waterfall model

***Requirment are must be fixed**

-> when to use

-> requirement are clear document fixed

-> product definition are fixed

-> project is short

<u>Pros</u>

. simple and easy to use

. Easy to arrange task

. Process and stages are clear

. Easy to manage due to rigidity

<u>Cons</u>

. Not good model for complex model

. High risk and you cannot requirement

Q-6 Write phases of spiral model

Spiral model is used for medium to hight it,s is used for when customer is not sure

Pros

. You can change requirement anytime

. Used see the system early

. Development can be divided into smaller part and requirement can be captured accurately

Cons

. Process is complex spiral

. Not stable for more project

. Management is more complex

Q-7 Write agile manifesto principles

. **Individuals and interactions –** in agile development, self-organization and motivation are

Important, as are interactions like co-location and pair programming.

. **Working software –** 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 factors, continuous customer interaction is very important to get Proper product requirements..

. **Responding to change –** agile development is focused on quick responses to change and Continuous development

Q-8 Explain working methodology Of agile model and also write pros and cons.

. Agile SDLC model is a combination of iterative and incremental . Process models with focus on process adaptability and customer . Satisfaction by rapid delivery of working software product.

. Agile Methods break the product into small incremental builds.

. These builds are provided in iterations.

. Each iteration typically lasts from about one to three weeks

Pros

. Minimal rules, documentation easily employed.

. Enables concurrent development and delivery within an

. Overall

. Planned ccontext

. Little or no planning required . Easy to manage

. Gives flexibility to developers

Cons

. Not suitable for handling complex dependencies.

. More risk of sustainability, maintainability and extensibility. **.** An overall plan, an agile leader and agile PM practice is a must without

Which it will not work.

. Strict delivery management dictates the scope, functionality to be

Delivered, and djustments to meet the deadlines

Q -9 what is SRS

. A software requirements specification (SRS) is a complete description Of the behavior of the system to be developed.

. It includes a set of use cases that describe all of the interactions that The users will have with the software.

. Use cases are also known as functional requirements. In addition to use

Cases, the SRS also contains nonfunctional (or supplementary) Requirements.

. Non-functional requirements are requirements which impose

Constraints on the design or implementation (such as Performance requirements, quality standards, or design Constraints).

. Recommended approaches for the specification of software

Requirements are described by IEEE 830-1998.

. This standard describes possible structures, desirable contents, and

Qualities of a software requirements specification

Q-10 What is oops

. Programming is like writing.

. If you can write a, demonstration You can make a program.

. So, programming is also easy.

. But, actually, programming is not so Easy, because a real good program is Not easily programmed. It needs the Programmers' lots of wisdom, lots of Knowledge about programming and Lots of experience.

. It is like writing, to be a good writer Needs lots of experience and lots of

Knowledge about the world.

. Learning and practise is necessary

Q-11 Write Basic Concepts Of oops

. Object

. Class

. Encapsulation

. Inheritance

. Polymorphism

. Overriding

. Overloading

. Abstraction

Q- 12 What is object

. Tangible Things	As a car, printer, …
. Roles	As employee, boss, …
. Incidents	As flight, overflow, …
. Interactions	As contract, sale, …
. Specifications	As colour, shape, …

Q-13 What is class

A class represents an abstraction of the object and abstracts the Properties and behavior of that object

Q-14 What is encapsulation

Encapsulation is the practice of including in an object everything

It needs hidden from other objects. The internal state is usually

Not accessible by other objects.

Q-15 What is inheritance

Inheritance means that one class inherits the characteristics of

Another class. This is also called a “is a” relationship

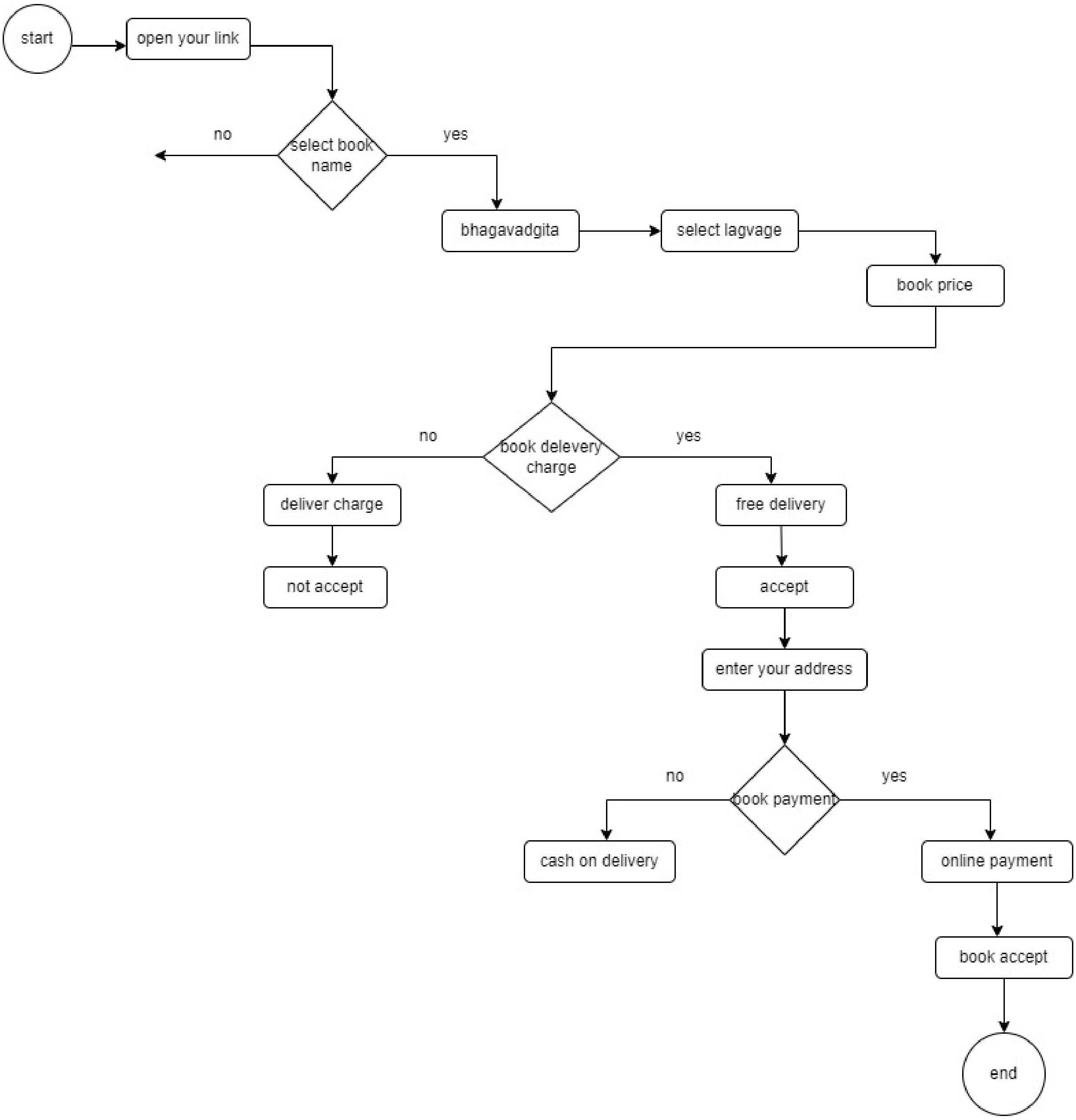
Q-16 What is polymorphism

. Polymorphism means “having many forms”.

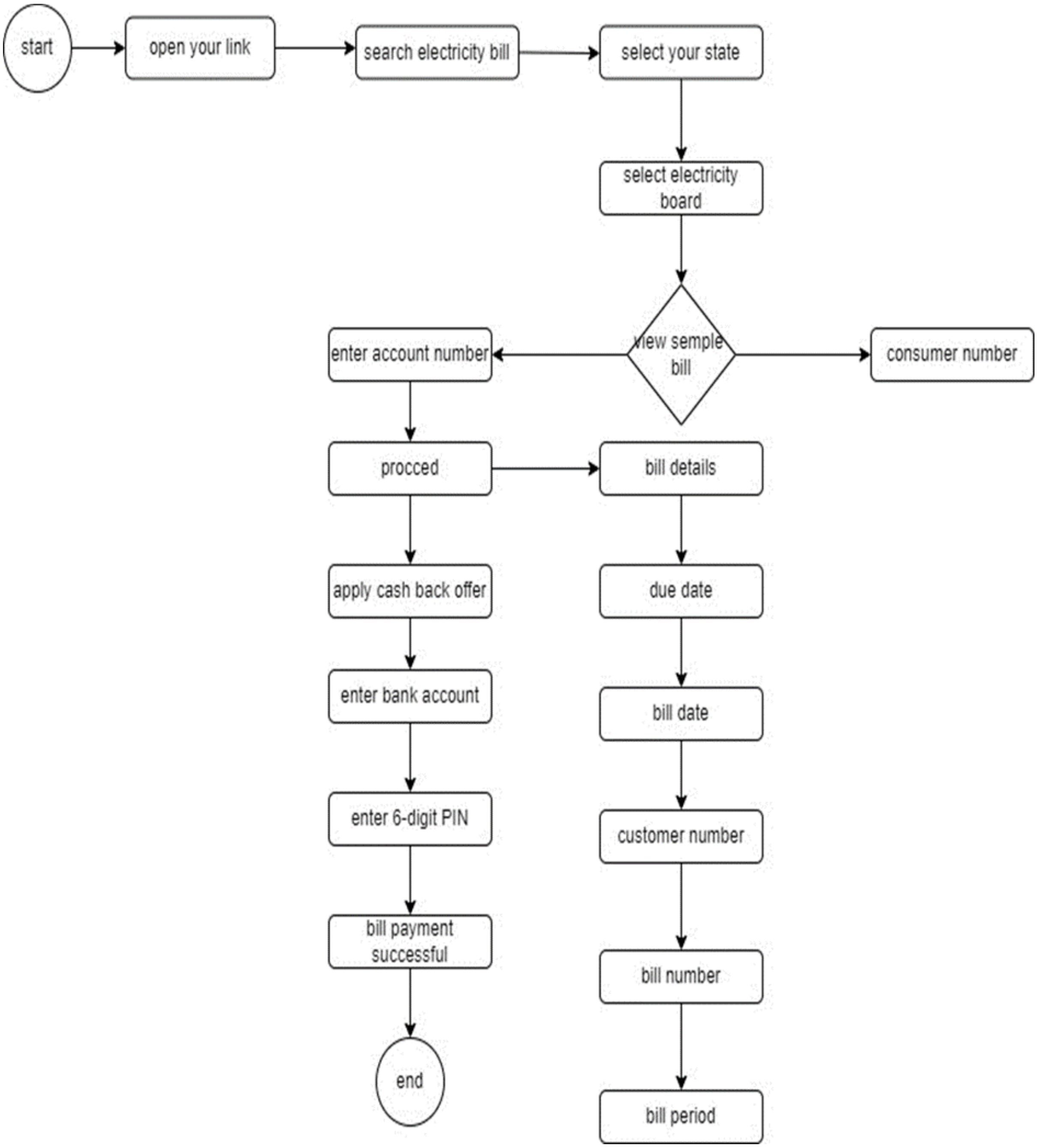
. It allows different objects to respond to the same message in different

Ways, the response specific to the type of the object

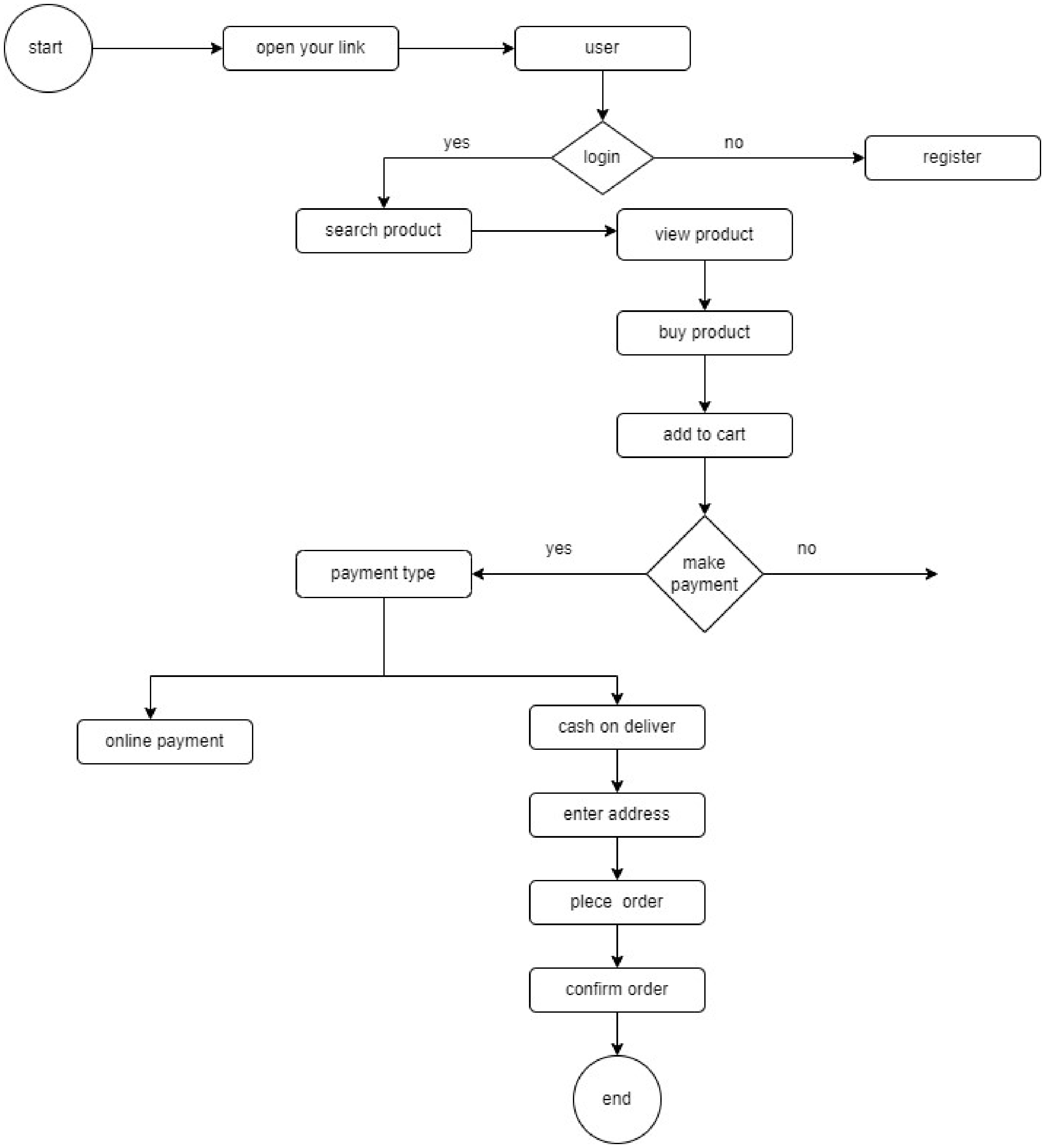
Q-17 Draw Usecase On Online book shopping


start
open your link
select book name
no
yes
bhagavadgita
select lagvage
book price
book delevery charge
no
yes
deliver charge
not accept
free delivery
accept
enter your address
book payment
no
yes
cash on delivery
online payment
book accept
end

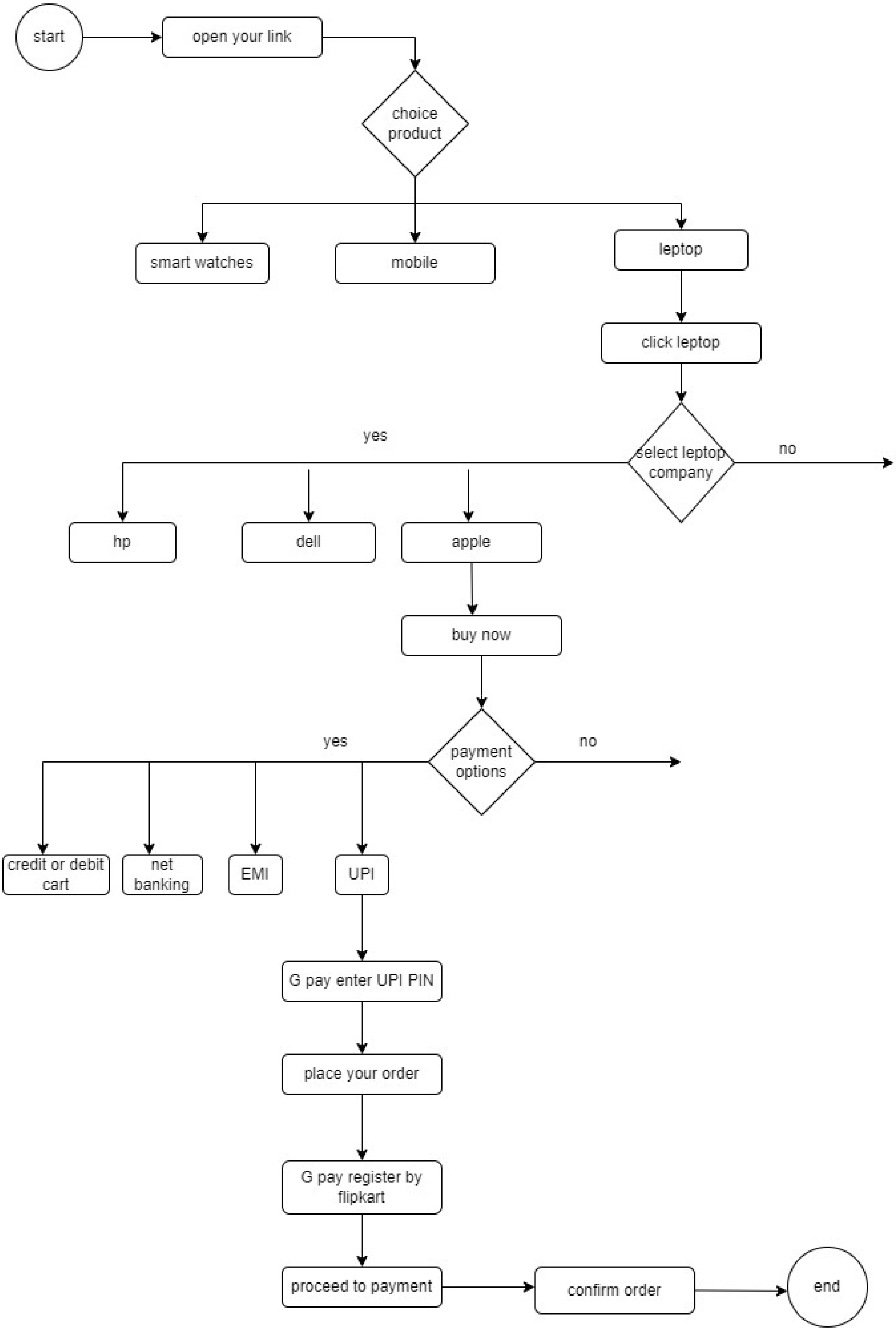
Q-18 DRAW USECASE ON ONLINE BILL PAYMENT SYSTEM (PAYTM)



Q-19 DRAW USECASE ON ONLINE SHOPPING PRODUCT USING COD.



Q-20 DRAW USECASE ON ONLINE SHOPPING PRODUCT USING PAYMENT GATEWAY.


start
open your link
choice product
smart watches
mobile
leptop
click leptop
yes
select leptop company
no
hp
dell
apple
buy now
yes
payment options
no
credit or debit cart
net banking
EMI
UPI
G pay enter UPI PIN
place your order
G pay register by flipkart
proceed to payment
confirm order
end