Q1) What is software? What is software engineering?

A1) Software is the set of instructions, data or program used to operate computers and execute specific tasks.

It is the opposite of the hardware which describes the physical aspects of the computer.

software is a generic term used to refer to applications scripts and programs that run on a device.

what is software engineering?

software engineering is a detailed study of engineering to the design development and maintenance of the software.

software engineering was introduced to address the issues of low quality software projects.

Problem arises when software generally exceeds timelines budget and reduced levels of quality.

Q2) Explain types of software?

A2) There are 4 main types of software

1) Application software

2) System software

3) Programming software

4) Driver software

Application software :- Application software is designed for end users.

System software :- System software is designed for computer or mobile devices.

Programming software :- Programming software is for computer programmers and developers who are writing code.

Driver software :- Driver software is a set of files that tells a piece of hardware how to function by communicating with operating system.

Q3) What is SDLC? Explain each phase of SDLC?

A3) SDLC full from Software development life cycle is a structured process that enables the production of high quality low cost software in the shortest possible production time.

Phase of sdlc?

Ans:- There are 7 phases of SDLC

1) Planning
2) Analysis
3) Design
4) Coding and implementation
5) Testing
6) Deployment
7) Maintenance

1) Planning :- This **phase of the software development life cycle** sounds obvious, but it is worth exploring. Without good planning, a project will not have a clear scope and purpose. The goals, costs, and teams' structure are set up at this stage. Furthermore, during the planning (and every later stage), there is also room for constant feedback from the target group, developers, and other stakeholders.

2) Analysis :- This is the time to think about precisely what requirements the application should meet. At this phase, the developers often create a software requirements specification document. It also includes the functionalities that the application should offer. With requirements clearly defined, the design of the UI & UX of the application and its other elements like front- and backend, API or 3rd party services will be more accessible.

3) Design :- The team will be focused on the application architecture and programming, including defining the application's programming language/industry practices and methods of solving problems and performing specific tasks. The team begins to create the user interface and choose the platforms on which the software will run based on outlined application designs. And last but not least, security. How will the software be protected? How will user passwords and data be secured? This is the stage in which to tackle these issues. When the design and prototype are done, it is time for coding and implementation

4) Coding and implementation:-T he developers' work gets up to speed when it comes to the coding phase. Every feature designed earlier needs to be changed into code, and all components have to be implemented. If there is more than one developer working on the project (and that is the most common scenario) a focus on teamwork is also needed. A further priority is to find and fix bugs and errors as soon as possible in order to deploy high-quality code. To make the developers' job easier, it is worth preparing detailed documentation as a guide to better understand the application's aim and purpose.

5) Testing:- This stage is completed before releasing the product to users or starts even before coding in test-driven development (TDD). Most tests (if not all) should be automated, especially if you have implemented CI/CD pipeline. The goal of the testing phase is to ensure that every feature works as expected.

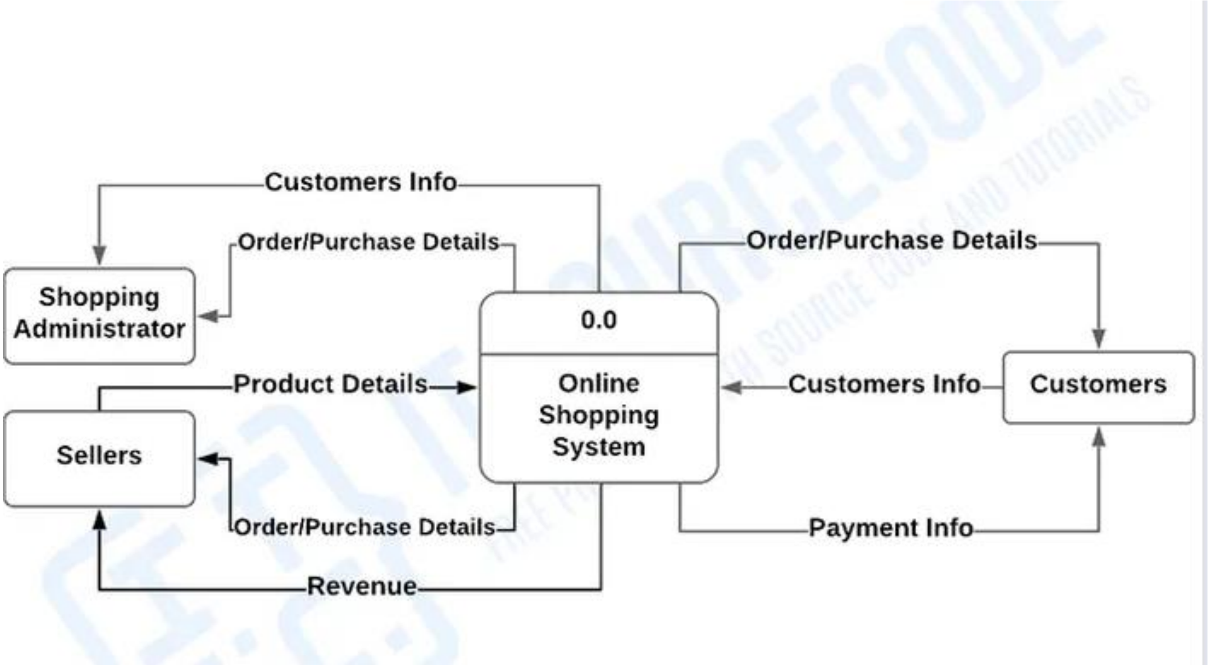
6) Deployment :- When the testing achieves positive results the application is allowed to see the light of day and make it available to users or customers. This is a key moment to improve scenarios based on real-world situations. Even though this process is automated (as a general rule), you and your teams should stay watchful since deployment is a complex process. Often, several systems and devices must be integrated and in some cases, more time and effort can be necessary to complete this stage successfully.

7) Maintenance:- The maintenance stage is probably the most crucial one of the **SDLC process**. Based on users' feedback after using the product in a real environment, you are able to improve your product with new features and eliminate any recurring bugs and possible vulnerabilities. The role of the development team at this phase is to look after the existing product, keeping it up-to-date with modern user needs and technology requirements.

Q-4) What is DFD? Create a DFD diagram on Flipkart

A4) A data flow diagram is a graphical or visual representation using a standardized set of symbols and notations to describe a business operation through data movement.

DATA FLOW DIAGRAM



Q-5) What is flowchart? Create a flowchart to make addition of two numbers

A5) Flowchart are nothing but the graphical representation of the data or the algorithm

For the better understanding of the code visually. It display step by step problem or process .

A flowchart is a picture of boxes that indicates the process flow in a sequential manner.

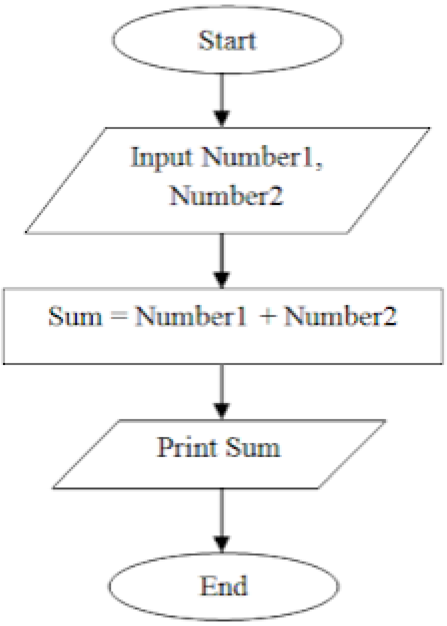
Use of flowchart

- It is a pictorial representation of an algorithm that increases the readability of the program.
- Complex programs can be drawn in a simple way using a flowchart.

Algorithm:

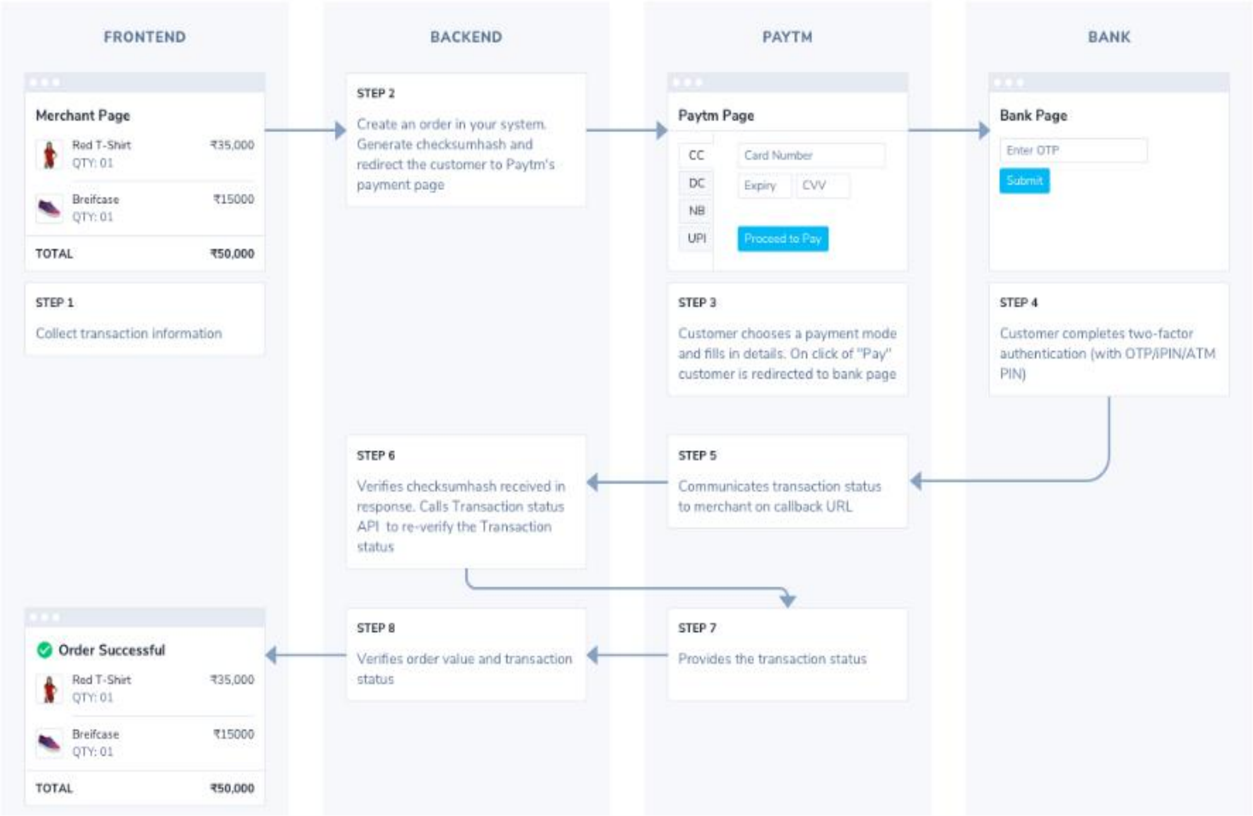
```
Step 1: Start
Step 2: Declare variables num1, num2 and sum.
Step 3: Read values for num1, num2.
Step 4: Add num1 and num2 and assign the result to a variable sum.
Step 5: Print sum
Step 6: Stop
```

Flowchart to Add two numbers



Q-6) What is use case diagram? Create a use case on bill payment on paytm.

A6) A use case diagram is used to represent the dynamic behaviour of the system. It models the tasks services and functions required by a system of an application.



STEP 1

Collect transaction information

STEP 2

Create an order in your system. Generate checksumhash and redirect the customer to Paytm's payment page

STEP 3

Customer chooses a payment mode and fills in details. On click of "Pay" customer is redirected to bank page

STEP 4

Customer completes two-factor authentication (with OTP/IPIN/ATM PIN)

STEP 5

Communicates transaction status to merchant on callback URL

STEP 6

Verifies checksumhash received in response. Calls Transaction status API to re-verify the Transaction status

STEP 7

Provides the transaction status

STEP 8

Verifies order value and transaction status