Modual-1(Fundamental)

* What is SDLC?

SDLC is a structure imposed on development of software product that defines the process for planning

, implementation,testing, deployment, documentation,and maintenance.

* What is software testing?

Software testing is a process used to identify the correctness,completeness,and quality of developed computer software.

* What is agile methodology?

Agile SDLC model is a combination of iterative and incremental process model 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.

* What is SRS?

A software requirement specification is a complete description of the behaviour of the system to be developed.It includes a set of use cases that describe all of the interactions that the user will have with the software.

* What is oops?

Object oriented programming systems.

Blackbox testing, functional testing.

* Write the basic concepts of oops.

There are 6 basic concepts of oops

1. Class
2. Object
3. Encapsulation
4. Inheritance
5. Polymorphism
6. Abstraction

* What is object?

Instances of a class.

To create a memory of a class.

To access the whole properties of a class except private.

* What is a class?

Class is a collection of data member(variable)and member function (process, methods) with its behaviours.

* What is Encapsulation?

Encapsulation means wrapping up of data into single unit private your data member or member function.

* What is inheritance?

Inheritance means properties of parent class extends into child class main purpose is reusability, extensibility.

Five type of inheritance

1. Single
2. Multilevel
3. Hierarchical
4. Multiple (Java does not support)
5. Hybrid (Java does not support)

* What is Polymorphism?

Polymorphism means ability to take one name having different forms, many forms or multiple forms.

Two type of polymorphism

1. Compile time (Method overloading)
2. Run time (Method overriding)

* Draw usecase on online book shopping.

1. Select the online book shopping app
2. Select the book (product)
3. Add to cart
4. Complete address process
5. Select the payment option
6. Conform the order
7. Waiting for order conformation screen

* Draw usecase on online bill payment system (Paytm)

1. Open Paytm app
2. Select bill payment option
3. Select company of electricity
4. Enter your customer Number
5. Enter your bill amount
6. Enter your Paytm transaction code (pin)
7. Wait for conformation screen

* Write SDLC phases with basic introduction.
* Requirement Gathering

1. Features.
2. Usage scenarios.
3. Although requirements may be documented in written form, they may be incomplete, unambiguous, or even incorrect.
4. Requirements will change!

* Analysis phase

1. The analysis phase defines the requirements of the system, independent of how these requirements will be accomplished.
2. The deliverable result at the end of this phase is a requirement document.
3. The deliverable design document is the architecture.

* Design phase

1. Design architecture document.
2. Implementation plan
3. Critical priority analysis.
4. Performance analysis.
5. Test plan.

* Implementation phase

1. In the implementation phase the team builds the components either document from the design phase and the requirement document from the analysis phase, the team should build exactly what has been requested though there is still room for innovation and flexibility.
2. The implementation phase deals with issues of quality, performance, baselines, libraries and debugging.
3. Critical error removal.

* Testing phase

1. Simply started quality is very important. Many companies have not learned that quality is important and deliver more claimed functionality but at a lower quality level.
2. A customer satisfied with the in software engineering and the process of enhancing and optimizing deployed software as well as fixing defects.
3. Updating all analysis, design, and user documentation.

* Maintenance phase

1. Maintenance is the process of changing a system after it has been deployed.
2. Corrective maintenance identifying and repairing defects.
3. Adaptive maintenance adapting the existing solution to the new platforms.
4. Perfective maintenance implementing the new on decides the utility and value of the software at a particular level of quality out weighs the impact of the known defects and deficiencies.

* Explain phases of the waterfall model.

1. Requirements collection
2. Analysis
3. Design
4. Implementation
5. Testing
6. Maintenance

* Pros

1. Simple and easy to use and understand.
2. Easy to manage due to the rigidity of the model.
3. Well understood milestone.
4. Easy to arrange tasks.
5. Process and results are well documented.

* Cons

1. No working software is produced until late during the life cycle.
2. High amount of risk and uncertainty.
3. Not a good model for complex and object-oriented projects.
4. Poor model for long and ongoing projects.

* Write phases of spiral model.

1. Planning: determination of objectives, alternatives and constraints.
2. Risk analysis: analysis of alternative and identification resolution of risks.
3. Engineering: development of the next level product.
4. Customer Evaluation: assessment of the results of engineering.

* Write agile manifesto principles.

There are 4 agile manifesto principles.

1. Individual interaction
2. Working software
3. Customer collaboration
4. Responding to change

* Explain working methodology of agile model and also write pros and cons.
* Agile model methodology

1. Agile SDLC model is a combination of iterative and incremental process model with focus on process adaptability and customer satisfaction by rapid delivery of working software product.
2. Agile methods break the products into small incremental builds.
3. Each iteration typically lasts from about one to three weeks.
4. Every iteration involves cross functional teams working simultaneously on various areas like planning, requirements analysis, design, coding, unit testing and acceptance testing.
5. At the end of the iteration a working product is displayed to the customer and important stakeholders.

* Agile model pros

1. Promotes team work and cross training.
2. Functionality can be developed rapidly and demonstrated.
3. Resource requirements are minimum.
4. Suitable for fixed or changing requirements.
5. Good model for environments that change steadily.

* Agile model cons

1. Not suitable for handling complex dependencies.
2. Strict delivery management dictates the scope, functionality to be delivered, and adjustments to meet the deadlines.
3. Depends heavily on customer interaction so if customer is not clear, team can be driven in wrong direction.
4. Transfer of technology to new team members may be quite challenging due to lack of documentation use – case.

* Draw usecase on online shopping product using COD.

1. Open the online shopping app
2. Select the product
3. Add to cart
4. Complete address process
5. Select payment option
6. Select COD payment option
7. Conform the order
8. Waiting for the order conformation screen.

* Draw usecase on online shopping product using gateway.

1. Select the product
2. Add to cart
3. Complete address process
4. Select payment option
5. Select debit card option
6. Enter card no and expiry date
7. Enter card cvv no
8. Enter OTP no
9. Wait for transaction complete mess