1. What is software testing?

Software testing is the process use for identify the correctness, completeness and quality of the computer developed software.

Software testing is used for finding the defect in the software.

1. What is 7 key principles? Explain in detail?  
   a) Testing shows presence of defects

b) Exhaustion is not possible in testing

c) Early testing

d) Defect clustering

e) The pesticide paradox

f) Testing is context dependant

g) Absence of errors fallacy

1. Testing shows presence of defects: software testing can shows the defect in software but cannot prove defect are not present.

Testing reduce the probability of undiscovered defect in the software but if defect is not found does not prove correctness.

Finding more defect reduce the probability of undiscovered defect.

1. Exhaustion is not possible in testing: if testing include all possible combinations of input stand tasting is not possible

So instead of doing exhausting testing focus on risks and priorities of testing.

1. Early testing: testing activities is start as early as possible so we can minimize the defect at early stage.

These activities focus on defined objectives.

1. Defect clustering: A small number of modules contain more defects discovered during testing.

Defects are not spread they are in clustered.

1. The pesticide paradox: if same test are repeated over and over test cases will no longer find new defects.

So to avoid pesticide paradox regularly reviewed and revised the test cases so we can find new defect and improved the software.

1. Testing is context dependant : testing is context dependant. Different type of sits tested differently. i.e safety software tested differently from ecommerce site.
2. Absence of errors fallacy:if the software is unusable and not full fill the requirement of costumer finding and fixing defects does not help.

3. What is SDLC?

SDLC: software development life cycle

SDLS is the structure imposed on development of software product that define as process of planning, implementation, testing, documentation, and mentation and support.

4. Write SDLC phases with basic introduction

SDLC Phases:

1) Requirements gathering

2) Analysis

3) design

4) implementation

5) testing

6) maintenance

1) Requirements gathering :

Featured

Useas scenario

Requirements may be in documented in written from, may be incomplete, unambiguous or incorrect

Requirements can be change in this phase

Testing is done in all the phase to prevent defec

Requirements can be functional and non functional

Functional requirements: this requirement is software function or service

Non functional requirements: this requirement is constrain or defec in software development. Non functional requirements is more critical than function requirement

During requirement gathering some problem can arise

A) lack of clarity: it is difficult to made documents both precious and easy to read

B) requirement confusion: functional and non functional requirements can be inttereind.

C) Requirements amalgamation: different requirements can be expressed together.

2) analysis phase:

This phase defines the requirement of system independent how to requirement will accomplish

This phase also define the problem that user try to solve

End product of this phase is requirement document

3) design:

Design architecture document

Implementation of plan

Design critical priority

Performance analysis

Test plan

4) implementation/coding:

In this phase team build the software by coding with the help of documents.

Given the architecture document from design phase and requirements document from analysis phase and build the exactly same as par the documentation.

5) testing phase:

This phase is most important phase but some company not learned the quality is important. With the testing increase the functionality of the product.

6) maintenance phase:

One of the activities of software engineering it is the process of enhancing and optimization of deployment software.

Maintenance work after deployment of software.

maintenance can be 3 types

A) corrective maintenance: identify the defect and raparing

B) adaptive maintenance: adapting existing solution to new platform.

C) perfective maintenance: implementation of new software.

5. What is agile methodology?

Agile is the combination of iterative and incremental model with focus on process adaptability and costumer satisfaction by radiply delivery of working software product.

6.What is SRS.

Software requirements specification is the complete description of the behaviour of the system to develop.

SRS contained use cases which describes all intarection use by users.

7. Explain Phases of the waterfall model

Waterfall model is classical software lifecycle model step by step waterfall between the various development phase.

Waterfall has six phases

A) requirements gathering: all requirements are collect at the initial phase. Requirements should be fix and clear

Onec software is done no changes in requirements

B) analysis

C) Design

D) implementation

E) Testing

F) Maintenance

8. Write phases of spiral model

Spiral model devided in to four phases

A) planning

B) risk analysis

C) engineering

D) customer evolution

A) Planning: in this phase team determine the objectives alternative and constraints.

Coustomer provide initial requirement of project.

All the requirements are collected from the user and plan according to the requirement.

B) risk analysis: after collecting all the requirements from the costume now analysis of risk factors start.

In this phase analysis of alternative and identification of risk factors is done

Risk of cost and delay in project is analysed, if cost is high than requirements may be change or completion of projects time is also change.

After complet analysis of risk factors now team will decide the project will go or will not going to next phase.

C) engineering:

If the project is enter in this phase than first prototype is done it's work as courier transfer the requirements.

Development of the next level product is start and evolving new system for development, coding implementation start as par initial requirement of customer.

D) customer evolution: after implantation of product it's deployed to customer. Customer take a alpha demo of product if they have any defect they inform the developer team and correct it.

In this phase assessment of results of engineering is processed.

9.Write agile manifesto principles.

Agile manifesto have four principals.

1) Individual intarection

2) working software

3) costumer collaboration

4) responding in to change

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

Agile is the combination of iterative and incremental model with focus on process adaptability and costumer satisfaction by radiply delivery of working software product.

This model believe that every project need to hendal differently. Model should be tailored to best suit project requirements.

this model divided in to time box for delivery specific features for release

Working software is build after each iteration and each iteration have incremental features, final build have all the features required by the customer.

Each iteration last arround one to three weeks

Agile thought process started early in the software development and became popular with time because of flexibility and adaptability.

Pros:

It is a realistic approach to software development

Promot the team work

Minimum documentation required

Functionality can be developed rapidly

Resources requirements minimum

No planning required easy to manage

Suitable for changing requirements

Cons:

Not suitable for complex dependency

More risk of sustainability, maintainability and extensibility

Depends on costumer communication, of costumer not clear team can be driven wrong direction.

Individual dependency so minimum documentation generated.

11. What is oops?

Oops is object oriented programming system

Identifying objects and assigning responsibility of objects

12. What is object?

Object is instance of an class.

Object represent individual items or unit.

13. What is class?

Class is collection of data members(variable) and member function(method or process) with its behaviour.

14. What is encapsulation?

Wrapping up of data into single unit. Private data members and data function.

15. What is inheritance?

Properties of parent class extend in to child class. Main purpose is reusability and extendsibility.

16.What is polymorphism?

Ability to take one name having different forms.

There are mainly two types

A) method overloading

B) method overriding

17. Write Basic Concepts of oops?

Oops have mainly six concepts

A) class

B) Object

C) polymorphism

D) inheritance

E) encapsulation

D) abstraction

