

Software engineering Interview Questions And Answers Guide.



Global Guideline.

<https://globalguideline.com/>



Software engineering Job Interview Preparation Guide.

Question # 1

Define Installation Testing?

Answer:-

System testing conducted once again according to hardware configuration requirements. Installation procedures may also be verified.

[Read More Answers.](#)

Question # 2

Define Unit Testing?

Answer:-

The Tools used in Unit Testing are debuggers, tracers and is Done by Programmers. Unit testing verifies the functioning in isolation of software pieces which are separately testable.

[Read More Answers.](#)

Question # 3

Define Stress testing?

Answer:-

We impose abnormal input to stress the capabilities of the software. Input data volume, input data rate, processing time, utilization of memory, etc. are tested beyond the designed capacity.

[Read More Answers.](#)

Question # 4

Define Functional Testing?

Answer:-

It checks that the functional specifications are correctly implemented. Can also check if Non Functional behavior is as per expectations.

[Read More Answers.](#)

Question # 5

Define User Acceptance Testing?

Answer:-

Testing conducted to determine whether or not a system satisfies its acceptance criteria and to enable the customer to determine whether or not to accept the system. It is Done against requirements and is done by actual users.

[Read More Answers.](#)

Question # 6

Define Performance Testing?

Answer:-

It is verifying that the software meets the specified performance requirements (response time, volume.

[Read More Answers.](#)

Question # 7

Define Configuration Testing?

Answer:-

It is Used when software meant for different types of users. It also check that whether the software performs for all users.

[Read More Answers.](#)



Question # 8

Define Usability Testing?

Answer:-

It evaluate the Human Computer Interface. Verifies for ease of use by end-users. Verifies ease of learning the software, including user documentation. Checks how effectively the software functions in supporting user tasks. Checks the ability to recover from user errors.

[Read More Answers.](#)

Question # 9

Define Security Testing?

Answer:-

It is used to verify proper controls have been designed.

[Read More Answers.](#)

Question # 10

Define Recovery Testing?

Answer:-

It is used in verifying software restart capabilities after a "disaster".

[Read More Answers.](#)

Question # 11

Define Sanity Test?

Answer:-

It is used to Verify whether the build is ready for feature/requirement based testing).

[Read More Answers.](#)

Question # 12

How you can Design a Test Case?

Answer:-

This Question is often asked in Microsoft or from any other company who is looking to hire you for testing work. A test case will have 5 sections.

[Read More Answers.](#)

Question # 13

Define Alpha / Beta testing?

Answer:-

Probably one term which you must be aware of as we often hear this software is in Alpha phase and in Beta phase. Here Testing is done by representative set of potential users for trial use. Please Note - in-house (alpha testing) - external (beta testing).

[Read More Answers.](#)

Question # 14

Define about Regression Testing?

Answer:-

Regression Testing according to IEEE is "selective retesting of a system or component to verify that modifications have not caused unintended effects". It is repetition of tests intended to show that the software's behavior is unchanged, except insofar as required. It can be done at each test level.

[Read More Answers.](#)

Question # 15

Define System Testing?

Answer:-

IEEE defines it as The process of testing an integrated hardware and software system to verify that the system meets its specified requirements. It is tested against system specification. May test manual procedures, restart and recovery, user interface, stress, performance. In System Testing, real data is used and sometimes users' participation is also used.

[Read More Answers.](#)

Question # 16

Define Integration Testing?

Answer:-

According to IEEE, Integration Testing is An orderly progression of testing in which software elements, hardware elements, or both are combined and tested, to evaluate their interactions, until the entire system has been integrated. It tests against system design and focuses on communication between modules. Start with one module, then add incrementally. Various Types of Integration Testing are: * big bang approach - Integrate Everything at once * top-down approach - Keep on breaking the system in parts one by one and then test each part. * bottom-up approach - Test the small parts first and then keep on integrating the system and keep on testing the bigger module of it. * mixed approach - Done by help of stubs.

[Read More Answers.](#)



Question # 17

What are the main steps involved in developing a software?

Answer:-

The steps involved are

Planning

Analysis

Design

coding

testing.

After testing maintenance of the product is also involved.

[Read More Answers.](#)

Question # 18

What is software engineering, anyway?

Answer:-

Some basic definitions of software engineering are:

The application of a disciplined engineering approach to the development of software systems.

A body of knowledge and experience in software development practice and process.

[Read More Answers.](#)

Question # 19

I like computers. Is Software Engineering for me?

Answer:-

Software Engineering is all about using engineering principles for the production of software. If you like computers, like solving challenging problems and would like to make an impact on the world in which we all live, you should consider software engineering (SE).

Computer systems are an integral part of today's society. Software is a critical component of all computer systems, including the "embedded systems" used in communication networks, vehicles, consumer electronics, and medical devices. Software engineers have the knowledge and skills needed to produce high-quality, effective software on which all these computer systems depend.

BSSE program is one of the first four ABET accredited programs in the United States. Our curriculum provides a strong foundation upon which to build a successful SE career.

[Read More Answers.](#)

Question # 20

What is the difference between computer engineering (CE), computer science (CS), and software engineering (SE)?

Answer:-

While computer engineering programs include courses in software, SE programs incorporate much more detail in software development practice and process, including advanced areas of software architecture, requirements management, quality assurance, and process improvement. Software engineering programs do not stress computer hardware and electronics as much as computer engineering programs do.

On the other hand, software engineering is based on computer science, as other engineering disciplines are based on natural or life sciences. However, software engineering adds an emphasis on issues of process, design, measurement, analysis and verification. In general, scientists seek new knowledge, while engineers want to build things, solve problems, and help people. Both roles are important.

[Read More Answers.](#)

Question # 21

Why are software engineers needed?

Answer:-

Computer systems are pervasive and have a major impact on society. Software is a critical component of all computer systems, including the "embedded systems" used in communication networks, vehicles, consumer electronics and medical devices. Software engineers have the knowledge and skills needed to produce high-quality, effective software on which all these computer systems depend.

[Read More Answers.](#)

Question # 22

I have no previous programming experience. Can I still be successful Software Engineer?

Answer:-

Any previous programming experience that you may have is certainly an asset, but it is not a requirement. It is a very common myth that SE is mainly programming. In fact, SE programs put a lot more emphasis on software development practice and process, including advanced areas of software architecture, requirements management, quality assurance and process improvement.

[Read More Answers.](#)

Question # 23

During the interview, do not obsess over little mistakes that happen.

Answer:-

On more than one occasion, when I gave a star candidate a coding question, he zeroed in on the most optimally performant solution, identified the boundary cases, and began writing well-designed code. Midway through the problem, he makes a little error -- getting the order of operations wrong on the first try, or having an off-by-1 error, or forgetting to declare a variable.

When I point it out, the candidate responds with horror and then becomes so nervous that it impacts his performance during the rest of the interview.

The fear is unfounded. An awesome candidate making a little error is like a concert violinist playing a challenging Brahms concerto and hitting two wrong notes.



Sure, the audience could tell that he made mistakes, but they don't get confused as to whether he's actually at Twinkle-Twinkle-Little-Star level. Even if you completely bomb one question, many interviewers ask you multiple questions and will forgive a single mishap. Even bombing an entire interview is recoverable if the other interviews go well.

[Read More Answers.](#)

Question # 24

What are the Different types of Architectures in Software Engineering?

Answer:-

An architecture expresses a fundamental structural organization schema for software systems. It provides a set of predefined subsystems, specifies their responsibilities, and includes rules and guidelines for organizing the relationships between them.

- * Object-Oriented/Abstract Data Style

- * Layered Hierarchies

- * Blackboard architecture

- involving shared access to data with loosely coupled agents

- * Client/Server

- * Three tier Client/Server architectures

- * Peer-to-Peer Architecture

[Read More Answers.](#)

Question # 25

What are use cases and class diagrams in Software Engineering?

Answer:-

Use cases are graphical representation of system parts and their interaction that is taking place. Normally we depict the part of the systems who are involved in some activity as actors.

Class Diagrams are a part of designing process. After coming up with use cases of the system, we take each use case and come up with the classes that we need to perform that functionality. Each class will have Class Name, Attributes and Operations associated with it.

Some of the features of the classes are -

- Each class should have well defined responsibilities

- These responsibilities should be cohesive i.e. the system as a whole looks sensible and looks together.

[Read More Answers.](#)

Question # 26

What are sequence diagram? What are package diagram? What are collaboration diagram?

Answer:-

Sequence Diagrams

They are pictorial representation of event happening according to the time line. In sequence Diagrams, we show respective class names by a small box and the event as an arrow.

Sequence Diagrams help in describing the normal course and alternative course of use cases.

VOPC - View of Participating Classes

VOPC is a combined representation (showing) of all classes on one page.

Collaboration Diagrams

- are an alternative presentation of a sequence diagram. We use boxes to describe objects, the lines connecting two boxes indicate that the objects collaborate with to one another and we use a multiplicity factor "*" to indicate that all elements of the aggregation receive a message.

Package Diagram

is complete set of sequence Diagrams or Collaboration Diagrams of the system.

[Read More Answers.](#)

Question # 27

Explain What are Design patterns?

Answer:-

Design Patterns are simple and elegant solutions of commonly occurring problems in software design. Design Patterns make it easier to reuse successful designs and architectures of experienced and Professional Designers.

There are three types of Patterns -

1. Creational - Concerned with creation of objects

2. Structural - Concerned with composition of classes or objects

3. Behavioral - characterize the ways in which classes and objects interact and distribute responsibility.

[Read More Answers.](#)

Question # 28

What are the characteristics of good design? Name some Design Tools?

Answer:-

Object Oriented Design Tools - Rational Rose, Rhapsody, Telelogic Tau G2

UI Design Tools - Rapid/Paper prototypes, Simple

SUMI (Software Usability Measurement Inventory) - It will tell you how your product compares to this standardization base: whether you are about average for the market, below, or above.

[Read More Answers.](#)

Question # 29

What is SDLC? What are the various SDLC models? Explain them?



Answer:-

What is SDLC

A software cycle deals with various parts and phases from planning to testing and deploying. All these activities are carried out in different ways, as per the needs. Each way is known as a Software Development Lifecycle Model (SDLC).

Some SDLC models

* The Linear model (Waterfall)

- Separate and distinct phases of specification and development

- All activities in linear fashion

- Next phase starts only when first one is complete

* Evolutionary development

- Specification and development are interleaved (Spiral, incremental, prototype based, Rapid Application development)

- Incremental Model (Waterfall in iteration)

- RAD (Rapid Application Development) - Focus is on developing quality product in less time

- Spiral Model - We start from smaller module and keeps on building it like a spiral. It is also called Component based development.

* Formal systems development

- A mathematical system model is formally transformed to an implementation

* Agile Methods

- Inducing flexibility into development

* Reuse-based development

- The system is assembled from existing components

[Read More Answers.](#)

Question # 30

What is RUP and PSP?

Answer:-

RUP stands for Rational Unified Process

It deals with -

• Iterative software development process.

• Visual Modeling of Systems

• Quality Management

• Change Control Management

• Deals with the role, the activity, and the artifact

• For managing OO Software Development

Some Features of RUP are -

• Online Repository of Process Information and Description

• Templates for all major artifacts, including:

• Requirements tracking (RequisitePro templates)

• Use Cases (Word Templates)

• Project Management (Project Templates)

• Process Manuals describing key processes

PSP stands for Personal Software Process

PSP Objectives are -

• To introduce individuals to a process-based approach to developing software

• To show individuals how to measure, estimate, schedule, and track their work

• To show individuals how to improve the quality of their programs

• In general PSP improves quality and productivity

- The time saved in testing because of better quality reduces time across the project by 20-40%

Advantage of PSP -

• To manage your work & assess/build your talents/skills

• To plan better

• To track your performance precisely

• To measure the quality of your software products

[Read More Answers.](#)

Question # 31

What is Testing in Software Engineering?

Answer:-

What is Testing in Software Engineering

Testing is running the program(or product) under various circumstances and conditions to find errors and bugs in it. This is important as releasing a faulty product will not only cause serious problems to the end user, it will also harm the companies reputation. There are various kind of Testing conditions and which one to use depends on type of product.

Types of Testing in Software Engineering

As per Test Target, there is * Unit Testing * Integration Testing * System Testing As per Test Objective, there is * User Acceptance Testing * Installation Testing * Functional * Alpha / Beta testing * Regression * Performance * Stress * Usability * Configuration * Smoke (Sanity Test) Let's see them one by one -

[Read More Answers.](#)

Question # 32

What is SRS?

Answer:-

SRS stands for Software Requirement Specification.

It establishes the basis for agreement between customers and contractors or suppliers on what the software product is expected to do, as well as what it is not expected to do.

Some of the features of SRS are -

• It sets permits a rigorous assessment of requirements before design can begin.

• It sets the basis for software design, test, deployment, training etc. It also sets pre-requisite for a good design though it is not enough.



- It sets basis for software enhancement and maintenance.
- It sets Basis for Project plans like Scheduling and Estimation.

[Read More Answers.](#)

Question # 33

What are Functional and Non Functional Requirements in Software Engineering?

Answer:-

Functional Requirements are the Expected functionality or services from the system.

Non - Functional Requirements are System property and constraints. Now Constraints can be on requirements itself. Some examples are of Response time, 24X7 availability of the system etc.

[Read More Answers.](#)

Question # 34

What is Requirement Engineering?

Answer:-

Software Requirements are Descriptions and specifications of a system. A requirement is a condition or capability to which the system must conform and Define. A system is designed with certain functionality in mind and to come up with that functionality and its specification right at the beginning is Requirement Engineering. Requirements Engineering is Eliciting, organizing, and documenting the requirements of the system. The process of establishing the services that the customer requires from a system and the constraints under which it operates and is developed Software requirements should be

• Clear

• Complete

• Unambiguous

• Quantitative

[Read More Answers.](#)

Computer Programming Most Popular Interview Topics.

- 1 : [PHP Frequently Asked Interview Questions and Answers Guide.](#)
- 2 : [C++ Programming Frequently Asked Interview Questions and Answers Guide.](#)
- 3 : [C Programming Frequently Asked Interview Questions and Answers Guide.](#)
- 4 : [Cobol Frequently Asked Interview Questions and Answers Guide.](#)
- 5 : [Visual Basic \(VB\) Frequently Asked Interview Questions and Answers Guide.](#)
- 6 : [Socket Programming Frequently Asked Interview Questions and Answers Guide.](#)
- 7 : [Perl Programming Frequently Asked Interview Questions and Answers Guide.](#)
- 8 : [VBA Frequently Asked Interview Questions and Answers Guide.](#)
- 9 : [OOP Frequently Asked Interview Questions and Answers Guide.](#)
- 10 : [Python Frequently Asked Interview Questions and Answers Guide.](#)

About Global Guideline.

Global Guideline is a platform to develop your own skills with thousands of job interview questions and web tutorials for fresher's and experienced candidates. These interview questions and web tutorials will help you strengthen your technical skills, prepare for the interviews and quickly revise the concepts. Global Guideline invite you to unlock your potentials with thousands of [Interview Questions with Answers](#) and much more. Learn the most common technologies at Global Guideline. We will help you to explore the resources of the World Wide Web and develop your own skills from the basics to the advanced. Here you will learn anything quite easily and you will really enjoy while learning. Global Guideline will help you to become a professional and Expert, well prepared for the future.

* This PDF was generated from <https://GlobalGuideline.com> at **November 29th, 2023**

* If any answer or question is incorrect or inappropriate or you have correct answer or you found any problem in this document then don't hesitate feel free and [e-mail us](#) we will fix it.

You can follow us on FaceBook for latest Jobs, Updates and other interviews material.
www.facebook.com/InterviewQuestionsAnswers

Follow us on Twitter for latest Jobs and interview preparation guides
<https://twitter.com/InterviewGuide>

Best Of Luck.

Global Guideline Team
<https://GlobalGuideline.com>
Info@globalguideline.com