1. O que é um diagrama de estado UML?

Um diagrama de estado UML (Unified Modeling Language) é um tipo de diagrama usado para representar o comportamento dinâmico de um sistema, descrevendo as transições de um objeto entre diferentes estados ao longo do tempo. Ele mostra os estados possíveis de um objeto e as transições entre esses estados em resposta a eventos específicos. Esses diagramas são úteis para modelar o comportamento de sistemas complexos, destacando as mudanças de estado e os eventos que desencadeiam essas mudanças.

1. Quais são os elementos principais de um diagrama de estados?

Entrada saída

*Estado*: Representa uma condição específica na vida de um objeto, como "ativo", "inativo", "em espera", etc.

*Transição*: Mostra como um objeto passa de um estado para outro em resposta a um evento específico. Geralmente são definidas ou linhas direcionadas entre estados.

*Evento*: Uma ocorrência que desencadeia uma transição de um estado para outro, como um clique do mouse, um sinal recebido, um tempo expirado, etc.

*Ação*: Uma atividade associada a um estado ou transição, decrescente o que acontece quando um evento específico ocorre.

*Guarda*: Uma condição que deve ser verdadeira para que ocorra uma transição. Funciona como um planejamento para determinar se a transição pode ou não ser acionada.

*Estado Inicial*: Representa o estado inicial do objeto quando ele é criado ou quando uma máquina de estados é ativada.

1. O que é um estado em diagrama de Estado?

Vai mapear uma transição

Representa uma condição específica na vida de um objeto, como "ativo", "inativo", "em espera", etc.

1. Explique o que é uma transição em D.E (Diagrama de Estados).

Uma transição em um Diagrama de Estados (DE) representa a mudança de um estado para outro em resposta a um evento, mostrando como um objeto se move de um estado inicial para um estado final ao ocorrer um evento específico. Essa transição é representada por uma seta ou linha entre os estados no diagrama e pode incluir condições (guardas) que determinam se a transição ocorre quando o evento é acionado.

1. Explique o conceito de estado final no D.E.

O estado final em um Diagrama de Estados (DE) representa o término de um processo ou ciclo de vida de um objeto. Ele indica que não há mais atividades a serem realizadas e é representado por um símbolo de círculo necessário, indicando o fim de uma

sequência de estados ou comportamento do objeto. Este estado não possui transições de saída, marcando o encerramento do processo no diagrama.

1. Conceitue qualidade de software?

Preencher os requisitos solicitados pelo cliente, acessível,

1. O que é um modelo de qualidade de software?

Um modelo de qualidade de software é um conjunto organizado de práticas, diretrizes e critérios que são utilizados para avaliar, medir e garantir a qualidade de um software ao longo de seu ciclo de vida. Ele define padrões e métodos para garantir que o software seja confiável, seguro, eficiente e atenda às necessidades e expectativas dos usuários. Esses modelos ajudam a identificar defeitos, melhorar processos de desenvolvimento e garantir a qualidade do produto final.

1. O que é teste de software?

Vai garantir que foi testado e que esta de comum acordo, procura falhas

Teste de software é um processo para examinar e avaliar um sistema ou aplicativo, buscando identificar defeitos, garantir seu funcionamento correto e verificar se atende aos requisitos estabelecidos, garantindo a qualidade do produto final.

1. Quais são os principais objetos do teste de software?

*Performance:*

*Garantir que a aplicação esta funcionando. – Garantir a qualidade*

*Identificar Defeitos*: Encontrar e reportar falhas, erros ou problemas no software para que sejam corrigidos.

*Garantir a Qualidade:* Verifique se o software atende aos requisitos e padrões de qualidade estabelecidos.

*Validar Funcionalidades:* Teste se todas as funcionalidades do software funcionam conforme o esperado.

*Garantir Confiabilidade:* Verifica a estabilidade e confiabilidade do sistema em diferentes condições.

*Melhor desempenho*: Avaliar e melhorar o desempenho do software em termos de velocidade, resposta e eficiência.

*Confirmar Usabilidade*: Avaliar a facilidade de uso e experiência do usuário ao interagir com o software.

*Assegurar Segurança*: identificar e corrigir vulnerabilidades que possam comprometer a segurança do sistema.

1. Cite as principais atividades envolvidas no teste de software?

Ciclo de Vida,

Planejamento de Testes, Análise de Requisitos e Especificações, Desenvolvimento de Casos de Teste, Execução de Testes, Registro e Rastreamento de Defeitos, Análise e Relatórios, Testes de Regressão, Testes de Aceitação do Usuário, Testes de Desempenho e Segurança, Encerramento dos Testes.

1. O que é um caso de teste?

Vai desenvolver uma rotina para garantir os testes

Um caso de teste é um conjunto de instruções apresentadas que descrevem as condições, ações a serem realizadas e resultados esperados para verificar uma funcionalidade específica do software. Ele é usado para testar e validar se o sistema funciona conforme o esperado, identificando falhas ou problemas durante o processo de teste de software.

1. O que é um teste de unidade?

Testar uma parte do sistema - individualmente

Um teste de unidade é uma técnica de teste de software onde partes individuais, chamadas de unidades, como funções, métodos ou classes, são testados de forma isolada para validar se cada unidade funciona corretamente. O objetivo é verificar se cada unidade produz os resultados esperados para entradas específicas, garantindo o funcionamento adequado de cada componente do software de maneira independente antes de integrá-los ao restante do sistema.

1. O que é teste automatizado?

Vai fazer uma rotina de teste automatizada, sem precisar testar manualmente a =s funções / rotinas

Teste automatizado é o processo de utilização de ferramentas e scripts para executar testes de software de forma automatizada, sem intervenção manual constante. Esses testes são criados e executados por meio de softwares e scripts específicos, permitindo a realização de testes repetíveis, rápidos e sistemáticos, aumentando a eficiência e a cobertura dos testes ao longo do ciclo de vida de desenvolvimento de software.

1. É possível ter testes unitários automatizados. Explique

Sim, é possível automatizar testes unitários. São testes automatizados que verificam automaticamente partes isoladas de código, como funções ou métodos, sem intervenção manual, utilizando scripts ou ferramentas específicas para garantir o funcionamento correto das unidades de código.