**Introdução à Engenharia de Software**

**Atividades de Aprendizagem e Avaliação**

**Nome: João Vitor Garcia RA:2160641**

Use esta cor em seu texto

* Considerando o conteúdo no link “Engenharia de Software – Preambulo”, complete
* Engenharia de Software é uma área da Computação dedicada a a propor e aplicar princípios de engenharia na construção de software.
* Engenharia de Software trata da aplicação de abordagens sistemáticas, disciplinadas e quantificáveis para desenvolver, operar manter e evoluir software.
* A Engenharia de Software surgiu da necessidade de que software fosse construído com base em princípios práticos e teóricos.
* Segundo Brooks existem dois tipos de dificuldades em Desenvolvimento de Software
* dificuldades essenciais – relacionadas à área do problema
* dificuldades acidentais – relacionadas à tecnologia
* São dificuldades essenciais: a complexidade.; a conformidade; a facilidade de mudanças; e a Invisibilidade
* As 12 áreas de Engenharia de Software são:
* engenharia de requisitos
* projeto de software
* construção de software
* testes de software
* manutenção de software
* gerência de configuração
* gerência de projetos
* processos de software
* modelos de software
* qualidade de software
* prática profissional
* aspectos econômicos
* Os requisitos funcionais de um sistema definem o que um sistema deve fazer, isto é, quais funcionalidades ou serviços ele deve implementar.
* Os requisitos não funcionais de um sistema definem como um sistema deve oeprar, sob quais restrições e com qual qualidade de serviço.
* O projeto de um Sistema de Software se inicia pela definição da unidade de códigos, porém só no nível de interfaces (módulos)
* **Interfaces providas** se relacionam com o resto do sistema
* **Interfaces requeridas** se relacionam com unidades de código.
* A **Arquitetura de Software** trata da .organização de um sistema em um nível de abstração mais alto do que aquele que envolve classes ou construções semelheantes.
* Testes de software mostram a presença de bugs mas não a sua ausência
* Testes de usabilidade objetivam a usabilidade da interface do sistema.
* Os **testes** podem ser usados para **verificação** com o objetivo de verificar. ou para **validação** com o objetivo de garantir que um sistema atende à sua especificação.
* **Defeitos** são bugs já **falhas** ocorrem quando um código com defeito for executado
* Nem todo defeito resulta em uma falha pois pode acontecer que o código defeituoso nunca seja executado.
* O defeito no código do foguete ‘Ariane 5’ estava relacionado com a conversão de um número real, em ponto flutuante, com 64 bits, para um número inteiro 16 bits, a falha ocorreu quando um número maior do que o maior inteiro que pode ser representado cem 16 bits.
* As manutenções de software podem ser classificadas em: corretiva, preventiva, adaptativa, refactoring e evolutiva.
* Manutenção adaptativa tem por objetivo adaptar um sistema a uma mudança em seu ambiente, incluindo a tecnologia, legislação, regras de integração com outros sistemas ou demandas de novos clientes.
* *Refactoring* é um tipo de manutenção que tem por objetivo a melhoria de seu código ou projeto.
* Gerência de Configuração se relaciona com o conjunto de políticas para gerenciar as diversas versões de um sistema.
* A **Lei de Brooks** é: "A inclusão de novos desenvolvedores em um projeto que está atrasado contribui para torná-lo ainda mais atrasado".
* A gerência de Projetos se ocupa de atividades tais como: ; , ; prazos,contratos,Lei de Brooks,etc.
* .Um PROCESSO DE DESENVOLVIMENTO DE SOFTWARE define quais atividades e etapas devem ser seguidas para construir ou entregar um sistema de software. .
* Processos ***Waterfall*** (em cascata) foram inspirados nos processos usados em engenharias tradicionais e são largamente sequenciais.
* As etapas de um processo de software em cascata são:
* levantamento de requisitos
* análise
* projeto detalhado
* codificação
* testes
* implementação.
* O **Manifesto Ágil** foi produzido em ..fevereiro no ano de 2001 por um grupo de 17 engenheiros
* A principal característica de um PROCESSO DE DESENVOLVIMENTO DE SOFTWARE ÁGIL é que um sistema deve ser construído de forma incremental e iterativa.
* XP, Scrum., Kanban e Lean Development são exemplos de processos ágeis.
* A **Integração Contínua** recomenda que desenvolvedores integrem o código que produzem imediatamente, se possível todo dia.
* Modelos criados para entender um sistema já implementado são instrumentos de Engenharia Reversa
* A **Qualidade de Software** pode ser avaliada em duas dimensões:
* qualidade externa
* qualidade interna
* A qualidade externa considera fatores que podem ser aferidos sem analisar o código.
* Conceitue
* Robustez o software continua funcionando mesmo quando ocorrem eventos anormais.
* Eficiência O software faz bom uso de recursos computacionais.
* A qualidade interna considera propriedades e caracteristicas relacionadas com a implmentação de um sistema.
* São exemplos de atributos da qualidade interna
* Modularidade
* Legibilidade
* Manutenibilidade
* Testabilidade
* Cite um exemplo de métrica de processo: o numero de defeitos reportados em produção por usuários finais em um certo intervalo de tempo.
* Revisões de código tem por objetivo detectar bugs antecipadamente e disseminar boas práticas de engenharia de software entre os membros de um time de desenvolvimento.
* ***Over-engineering*** é o uso de recursos mais sofisticados em um contexto que não demanda tanta preocupação.