5409 – Engenharia de Software

20241018 – Luis Cardoso

Atividade7 – Faça um breve resumo do que são cada uma destas caracteristicas

Entregue o trabalho num ficheiro com o nome **atividade7 e o seu primeiro e ultimo nome**

**Faça uma breve analise dos temas dados na sessão**

Ferramentas de suporte ao processo

As **Ferramentas de Suporte ao Processo** são fundamentais para otimizar as várias etapas do desenvolvimento de software, garantindo que o processo seja eficiente, colaborativo e controlado. Estas ferramentas auxiliam em diferentes aspectos do ciclo de vida de desenvolvimento, desde a codificação até a entrega do produto.

1. **Controle de Versão**: Ferramentas de controle de versão, como Git, são essenciais para gerenciar alterações no código-fonte, permitindo que várias versões sejam acompanhadas e revertidas quando necessário. Elas facilitam a colaboração entre equipes de desenvolvimento, ao fornecer um histórico detalhado das modificações realizadas no projeto e permitir que múltiplos desenvolvedores trabalhem simultaneamente.
2. **Integração Contínua/Entrega Contínua (CI/CD)**: Ferramentas de CI/CD, como Jenkins e GitLab CI, automatizam a compilação, teste e entrega de software. Elas ajudam a detectar erros mais cedo no processo de desenvolvimento, permitindo uma integração e entrega mais rápida e eficiente. A CI/CD promove a prática de atualizar frequentemente o código e validar essas atualizações, melhorando a qualidade e a velocidade de lançamento.
3. **Gerenciamento de Projetos**: Ferramentas como Jira e Trello auxiliam no planejamento e acompanhamento das tarefas do projeto. Elas permitem que as equipes definam prazos, monitorem o progresso, atribuam responsabilidades e gerenciem o backlog. Isso melhora a visibilidade e a comunicação dentro das equipes, promovendo um fluxo de trabalho organizado.
4. **Documentação**: Ferramentas como Confluence ou Markdown facilitam a criação e manutenção de documentação técnica e de usuário. Elas garantem que todas as informações sobre o desenvolvimento, como especificações de requisitos e guias de uso, estejam bem organizadas e acessíveis, promovendo a continuidade e o suporte ao longo do ciclo de vida do software.
5. **Ambiente de Desenvolvimento Integrado (IDE)**: IDEs como Visual Studio Code ou IntelliJ IDEA proporcionam um ambiente unificado onde os desenvolvedores podem escrever, compilar e depurar o código. Essas ferramentas geralmente possuem suporte a extensões que aumentam a produtividade, como linters, depuradores e integração com controle de versão.
6. **Teste Unitário**: Ferramentas de teste unitário, como JUnit ou NUnit, são usadas para garantir que os componentes individuais do software funcionem corretamente. Testes automatizados podem ser integrados com CI/CD para garantir que o código submetido atenda a critérios de qualidade, evitando a introdução de novos erros.

Essas ferramentas desempenham um papel crucial em garantir a qualidade do produto final e a eficiência do processo de desenvolvimento, facilitando a colaboração, automação e monitoramento contínuo.