Avalição Teórica

1 - Cite uma abordagem desenvolvimento ágil e explique-a.

Uma metodologia bastante popular e que tenho experiência prática é o Scrum. Um framework para desenvolvimento baseado em ciclos curtos de trabalho, passando pelo backlog, plannig, Sprint, review e retro, onde as equipes trabalham juntas de forma colaborativa para entregar incrementos de software funcionais ao final de cada ciclo. O que eu acho legal no Scrum é sua flexibilidade e adaptabilidade, permitindo que as equipes se ajustem e melhorem de maneira progressiva. Além disso, o foco na entrega de valor ao cliente de forma iterativa é algo que realmente faz sentido para mim, pois podemos ver resultados em curtos períodos de tempo.

2 - O que é CI/CD?

A Integração Contínua e Entrega Contínua, são práticas que podem acelerar o processo de desenvolvimento e melhorar a qualidade do software, já que estão presentes no desenvolvimento e, com a integração com git, são enviadas automaticamente no final de cada alteração. A Integração Contínua envolve a integração frequente do código fonte ao repositório central, com testes automatizados que ajudam a identificar erros e garantir a qualidade do código. Já a Entrega Contínua é uma abordagem que automatiza a implantação do software em ambientes de produção, permitindo a entrega rápida e confiável de novas funcionalidades ou correções.

3 - O que é TDD, BDD e ATDD e quando são aplicadas?

São práticas aplicadas durante o desenvolvimento de software para garantir a qualidade do código e facilitar o seu entendimento. No TDD os testes são escritos antes mesmo da implementação do código, com o objetivo de validar o comportamento esperado. No BDD os testes são escritos em linguagem natural, focando no comportamento do sistema em termos de cenários de negócio. Já no ATDD os testes são escritos em colaboração entre desenvolvedores, testadores e stakeholders, é a integração do código com o BDD, com o objetivo de definir os requisitos do sistema. Atualmente, é comum que os testadores também desenvolvam essas práticas visando a garantia de qualidade de seus testes.

4 - O que é TDD, BDD e ATDD e quando são aplicadas? Cite uma heuristica de testes

utilizada para testes de front e de back e explique-as.

Uma heurística de testes muito utilizada para testes de front-end é a de navegabilidade, que consiste na usabilidade do sistema. Usar o sistema com um olhar de usuário final, verificando um bom fluxo de navegação, que seja de forma intuitiva e eficiente, sem criar dupla interpretação. Já para back-end a heurística de integração utilizada, verificando se os componentes do sistema estão integrados corretamente e funcionam em conjunto.

5 - Cite ferramentas de testes automatizados para Desktop, Web, API ́s e mobile.

Atualmente, temos diversas ferramentas de teste automatizado que podem melhorar significativamente a qualidade do software. Para testes de aceitação em aplicações desktop, o Selenium é amplamente utilizado. Para testes em aplicações web, o Cypress é uma opção popular. Para testes de APIs em Java, o RestAssured é comumente utilizado, e para testes em aplicativos móveis, o Appium é uma ferramenta bastante utilizada. Cada uma dessas ferramentas oferece recursos específicos para automatizar testes em diferentes tipos de aplicações, proporcionando mais eficiência e confiabilidade nos processos de teste.

6 - O que é shift-left?

Shift-left é uma abordagem no desenvolvimento de software que envolve antecipar os testes e atividades de garantia de qualidade para fases iniciais do ciclo de desenvolvimento, como análise de requisitos e design. Isso significa incorporar a qualidade desde o início do processo, de forma orgânica e integrada, permitindo que a equipe identifique e corrija problemas antes que se tornem mais complexos e custosos de resolver.