

1. Como estruturar um software?

R: Primeiro passo é ter um bom entendimento dos requisitos do software, começando com uma visão de mais alto nível, sem ser muito técnico, após isso, mapeie os requisitos funcionais e leve em consideração requisitos não funcionais. A partir de agora pense em como seu software vai ser dividido, por exemplo, vai ser um monolito ou uma arquitetura de micro-serviços. A partir disso crie seu primeiro protótipo e conforme a preparação de requisitos comece a desenvolver. Existem diferentes formas e combinações para a estruturação de um software, dentre elas arquitetura MVC, monolitos, micro-serviços, front-end, back-end, será funcional, será orientado a objetos, entre outras. Como boa prática é essencial o versionamento do software, assim terá a possibilidade de restaurar alguma versão em caso de erros, por exemplo. Testes são de extrema importância para garantir uma entrega de qualidade, como testes funcionais, unitários, de desempenho, segurança, entre outros.

2. Quais são os componentes de software?

R: Um componente em um software, é um subconjunto de um software independente mas que pode ser usado por outros softwares ou partes do mesmo.

3. Quais são os tipos de arquitetura de software?

R: Existem diversos tipos de arquitetura das quais podem ser aplicadas a um software, e pode ser determinada de acordo com necessidades e requisitos. Dentre as mais famosas estas são alguns exemplos: Arquitetura client-server, contém 2 lados no software, cliente e servidor, o cliente é responsável pela camada de interface e interação com o usuário, já o servidor pode ser responsável por uma lógica, como um cálculo por exemplo, ou até mesmo salvar informações com nome e email de um usuário em um banco de dados. Outro exemplo é a arquitetura MVC, da qual é dividido o software em 3 partes, a primeira é a Model, ela é responsável por estabelecer interação com os dados do software, logo em seguida vem a View, que tem como função apresentar e receber dados dos usuários finais e por último a Controller, que é a camada intermediária entre as duas anteriores, recebendo dados da View, processando conforme a necessidade e entregando para a Model.

4. Quais são as 3 camadas de software?

R: É uma arquitetura de software da qual o organiza em três camadas, das quais são: Camada de apresentação, esta camada é responsável pela interface do usuário final, permitindo assim visualização e manipulação de dados e possíveis interações. Camada do aplicativo, esta parte do software é responsável pela lógica, tratamento e processamento das informações recebidas da camada anterior. Camada de dados, por fim esta é responsável pelo armazenamento dos dados provenientes da camada de aplicação.

5. Quais são os tipos de software?

R: Existem diversos tipos de software, dentre eles os mais utilizados são: Software de aplicativo, Software de programação, Software de sistema, entre outros.

6. O que é um Plano de Projeto de Software?

R: É um documento que abrange todos os aspectos de um projeto de desenvolvimento de software, desde seu início até sua conclusão. Serve de guia para equipes, detalhando objetivos, escopos, cronogramas, recursos, riscos, dentre outras informações.

7. Quais são os 3 principais padrões da arquitetura web?

R: Cliente Servidor, Arquitetura em camadas e Arquitetura Model-View-Controller (MVC).