Conceitos Avançados em Arquitetura de Sistemas

1)

Aula 1 – O Que é Arquitetura de Sistemas

Introdução

Atualmente, praticamente todas as atividades do nosso dia a dia utilizam algum tipo de sistema computacional. Quando realizamos uma compra online, acessamos uma rede social, utilizamos um aplicativo bancário ou consultamos notas em um sistema escolar, estamos interagindo com softwares que foram cuidadosamente planejados.

Por trás de cada sistema existe uma estrutura responsável por organizar seu funcionamento. Essa estrutura é chamada de Arquitetura de Sistemas.

O Que é Arquitetura de Sistemas?

Arquitetura de Sistemas é a forma como um software é organizado. Ela define como os componentes serão distribuídos, como os dados serão armazenados, como os módulos irão se comunicar e como o sistema poderá crescer futuramente.

Exemplos Reais

Netflix, Instagram, YouTube e sistemas escolares utilizam arquiteturas bem definidas para suportar milhões de acessos.

Benefícios

Organização, manutenção, segurança, escalabilidade e desempenho.

Atividade Prática

Escolha um sistema utilizado no seu dia a dia e descreva suas funcionalidades.

Exercícios

1. O que é arquitetura de sistemas?
2. Cite três benefícios.
3. Por que o planejamento é importante?
4. Cite dois exemplos de sistemas.
5. O que pode acontecer sem uma boa arquitetura?

Aula 2 – Componentes e Módulos de um Sistema

Introdução

Os sistemas modernos podem possuir milhares de funcionalidades. Para facilitar o desenvolvimento, eles são divididos em partes menores.

Conteúdo

Componentes são partes específicas responsáveis por determinada função. Módulos agrupam componentes relacionados.

Exemplos

Sistema escolar: módulo acadêmico, financeiro e administrativo.
Sistema hospitalar: pacientes, consultas, médicos e financeiro.

Vantagens

Organização, manutenção, reaproveitamento de código e trabalho em equipe.

Atividade Prática

Crie possíveis módulos para um sistema de mercado.

Exercícios

1. O que é um componente?
2. O que é um módulo?
3. Cite duas vantagens da modularização.
4. Dê um exemplo de módulo.
5. Qual a diferença entre módulo e componente?

Aula 3 – Acoplamento e Coesão

Introdução

Acoplamento e coesão são conceitos fundamentais para avaliar a qualidade de uma arquitetura.

Conteúdo

Acoplamento representa o grau de dependência entre módulos. Coesão representa o quanto as funcionalidades de um módulo estão relacionadas.

Exemplos
Módulo acadêmico contendo apenas notas e frequência possui alta coesão.
Módulo misturando estoque, pagamentos e notas possui baixa coesão.

Benefícios
Baixo acoplamento e alta coesão facilitam manutenção e evolução do sistema.

Atividade Prática
Analise exemplos de módulos e identifique situações de alta ou baixa coesão.

Exercícios
1. O que é acoplamento?
2. O que é coesão?
3. Qual a situação ideal?
4. Cite um exemplo de alta coesão.
5. Cite um exemplo de alto acoplamento.

Aula 4 – Escalabilidade de Sistemas

Introdução
Muitos sistemas começam pequenos e crescem rapidamente. Para acompanhar esse crescimento, a arquitetura precisa ser escalável.

Conteúdo
Escalabilidade é a capacidade de um sistema crescer sem perder desempenho.

Tipos
Escalabilidade Vertical: melhoria do hardware.
Escalabilidade Horizontal: adição de novos servidores.

Exemplos
YouTube, Netflix, Instagram e TikTok precisam ser altamente escaláveis.

Benefícios
Melhor desempenho, crescimento sustentável e menor ocorrência de falhas.

Atividade Prática
Pesquise um sistema famoso e explique como ele poderia crescer para atender mais usuários.

Exercícios
1. O que é escalabilidade?
2. O que é escalabilidade vertical?
3. O que é escalabilidade horizontal?
4. Cite um sistema escalável.
5. Por que a escalabilidade é importante?

Aula 5 – Introdução ao Desenvolvimento Ágil

Introdução
O desenvolvimento ágil surgiu para tornar a criação de software mais flexível e adaptável às mudanças.

Conteúdo
O Manifesto Ágil foi criado em 2001 e apresenta valores que priorizam pessoas, colaboração, software funcionando e adaptação às mudanças.

Métodos Ágeis
Scrum, Kanban e XP.

Exemplo
Desenvolvimento de um aplicativo de delivery dividido em entregas menores.

Benefícios
Entregas rápidas, produtividade e satisfação do cliente.

Atividade Prática
Liste três funcionalidades iniciais para um aplicativo de biblioteca.

Exercícios
1. O que é desenvolvimento ágil?

2. Quando surgiu o Manifesto Ágil?
3. Cite dois valores do Manifesto Ágil.
4. Cite dois métodos ágeis.
5. Quais os benefícios do desenvolvimento ágil?

Aula 6 – Os 12 Princípios Ágeis

Introdução
Após a criação do Manifesto Ágil, foram definidos 12 princípios que servem como guia para equipes de desenvolvimento de software. Esses princípios ajudam a transformar os valores do Manifesto Ágil em ações práticas.

Conteúdo
Os princípios defendem a entrega contínua de valor ao cliente, aceitação de mudanças, comunicação eficiente e melhoria constante dos processos. Também incentivam equipes motivadas e foco na simplicidade.

Exemplos
Empresas como Spotify, Netflix e Nubank utilizam práticas inspiradas nos princípios ágeis para evoluir seus sistemas constantemente.

Importância
Os princípios ajudam as equipes a manterem o foco nas necessidades dos usuários e na qualidade do produto.

Atividade Prática
Pesquise três dos 12 princípios ágeis e explique com suas palavras o significado de cada um.

Exercícios
1. Quantos princípios compõem o desenvolvimento ágil?
2. O que significa melhoria contínua?
3. Por que a comunicação é importante?
4. Como os princípios ajudam o cliente?
5. Cite uma vantagem da aplicação dos princípios ágeis.

Aula 7 – Introdução ao Scrum

Introdução
Scrum é um dos frameworks ágeis mais utilizados no mundo. Seu objetivo é organizar o desenvolvimento de software por meio de ciclos curtos chamados Sprints.

Conteúdo
No Scrum, o projeto é dividido em etapas menores. Cada Sprint possui duração definida e gera uma entrega funcional ao final.

Funcionamento
Planejamento, desenvolvimento, acompanhamento diário e avaliação dos resultados.

Exemplos
Sistema escolar:
Sprint 1 – Cadastro de alunos.
Sprint 2 – Cadastro de professores.
Sprint 3 – Lançamento de notas.
Sprint 4 – Relatórios.

Benefícios
Maior organização, entregas frequentes e facilidade para corrigir problemas.

Atividade Prática
Divida um projeto simples em três Sprints.

Exercícios
1. O que é Scrum?
2. O que é uma Sprint?
3. Qual o objetivo das entregas frequentes?
4. Cite uma característica do Scrum.
5. Por que o Scrum é tão utilizado?

Aula 8 – Papéis do Scrum

Introdução
O Scrum define funções específicas para organizar o trabalho da equipe e garantir o sucesso do projeto.

Conteúdo
Os três papéis principais são Product Owner, Scrum Master e Time de Desenvolvimento.

Product Owner
Responsável por representar os interesses do cliente e priorizar tarefas.

Scrum Master
Auxilia a equipe na aplicação do Scrum e remove obstáculos.

Time de Desenvolvimento
Equipe responsável pela construção do software.

Exemplo
Em um aplicativo de delivery, o Product Owner define prioridades, o Scrum Master coordena o processo e os desenvolvedores criam o sistema.

Atividade Prática
Identifique os papéis do Scrum em um projeto fictício.

Exercícios
1. Quem representa o cliente?
2. Qual a função do Scrum Master?
3. Quem desenvolve o sistema?
4. Por que os papéis são importantes?
5. Cite os três papéis do Scrum.

Aula 9 – Product Backlog

Introdução
O Product Backlog é uma lista organizada contendo tudo que precisa ser desenvolvido em um projeto.

Conteúdo

Nele são registradas funcionalidades, melhorias, correções e novas ideias. Os itens possuem prioridades definidas pelo Product Owner.

Histórias de Usuário
As necessidades dos usuários são registradas por meio de histórias de usuário.

Exemplo
"Como aluno, desejo visualizar minhas notas para acompanhar meu desempenho."

Benefícios
Organização, priorização e controle das atividades.

Atividade Prática
Crie três histórias de usuário para um sistema de biblioteca.

Exercícios
1. O que é Product Backlog?
2. Quem define prioridades?
3. O que são histórias de usuário?
4. Cite uma vantagem do Backlog.
5. O Backlog pode mudar durante o projeto?

Aula 10 – Sprint e Sprint Planning

Introdução
A Sprint é o período de desenvolvimento utilizado no Scrum. Antes de começar uma Sprint é realizado o Sprint Planning.

Conteúdo
Durante o Sprint Planning, a equipe escolhe as tarefas que serão executadas e define metas para o período.

Exemplo
Sprint de duas semanas:
- Cadastro de usuários
- Tela de login
- Recuperação de senha

Benefícios
Organização, previsibilidade, foco e acompanhamento do progresso.

Atividade Prática
Monte uma Sprint contendo três funcionalidades para um sistema de mercado.

Reflexão
Por que definir metas antes de iniciar o trabalho é importante?

Exercícios
1. O que é uma Sprint?
2. O que acontece durante o Sprint Planning?
3. Qual o objetivo do planejamento?
4. Cite uma vantagem das Sprints.
5. Como a equipe avalia os resultados?