

PROJETO, IMPLEMENTAÇÃO E TESTE DE SOFTWARE

Aula 02 - Projeto de Software

Prof. Fabricio Freire, MSc in Cybersecurity

METAS DE APRENDIZAGEM

- Introdução
- A fase Projeto de Software
- Conceitos básicos
- Qualidade do projeto
- Modelo do projeto
- Conclusão

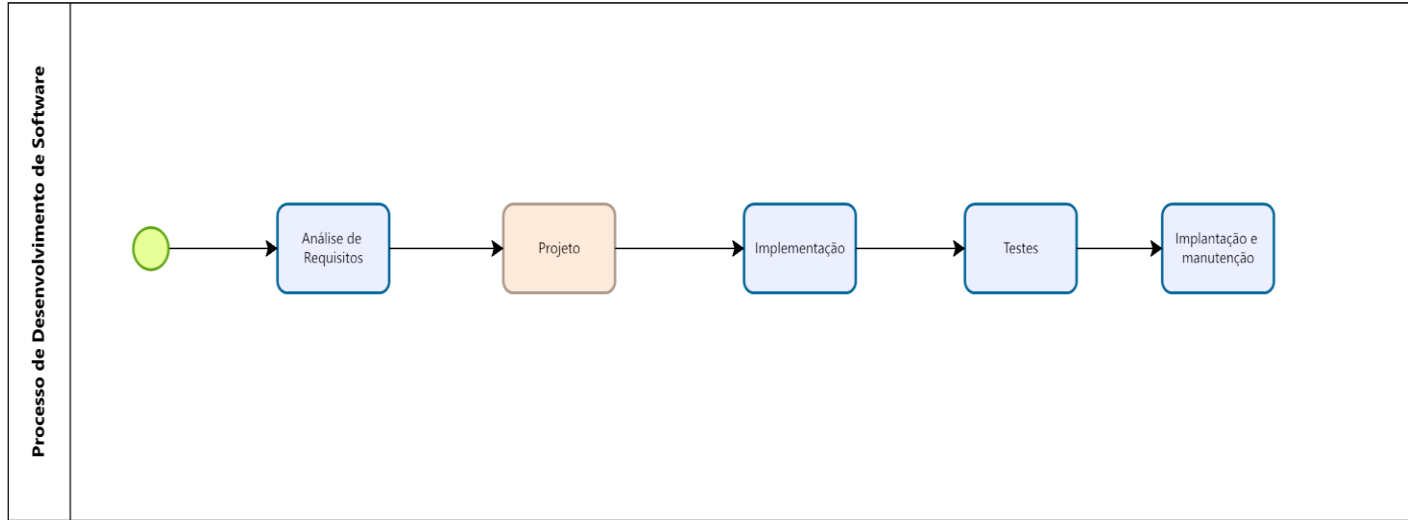
Engenharia de Software

Processo de Software

Etapas do processo de software



ONDE ESTAMOS???



PROJETO DE SOFTWARE

PROJECT

Mostrar e definir uma solução possível.

Como o software será construído

Arquitetura, a interface com o usuário, componentes



“Projeto é lugar onde a criatividade está em alta, em que os requisitos do cliente e a sua necessidade de sistema se juntam para o desenvolvimento de um produto ou sistema”

(Pressman, 2011)

FASE PROJETO DE SOFTWARE

Definição de uma estrutura que possa ser implementada baseada no que foi descrito nos requisitos que foram levantados para o sistema junto ao cliente.

Viabilidade

FASE PROJETO DE SOFTWARE

Análise de Requisitos	Projeto de Software
<ul style="list-style-type: none">• Domínio do problema• Analista	<ul style="list-style-type: none">• Solução para o problema• Comportamento do sistema• Engenheiro/Arquiteto

ARQUITETO DE SOFTWARE

- Experiência como programador
- Gerência de projetos
- Compreender e dominar as tecnologias envolvidas



ARQUITETO DE SOFTWARE

- Modelagem e metodologias
- Estratégias de negócios local
- Conhecimento sobre a concorrência



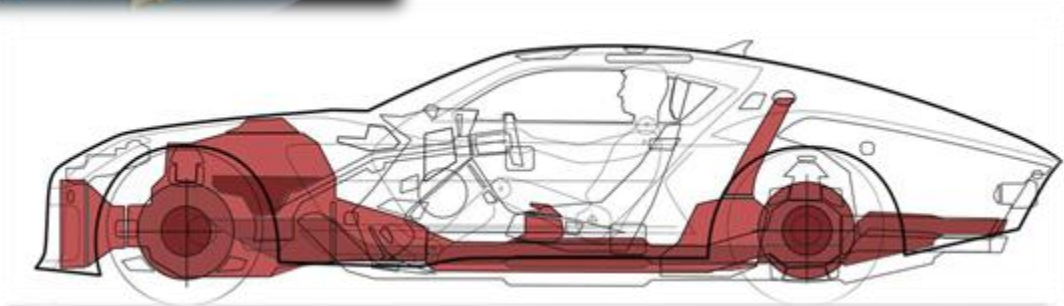
ARQUITETO DE SOFTWARE

- Modelagem e metodologias
- Estratégias de negócios local
- Conhecimento sobre a concorrência



A fase Projeto de Software

- Permite que o sistema seja modelado ou o produto construído.
- Avaliação para verificar a qualidade e com isso ser aperfeiçoado antes de passar para a fase de implementação



“...o projeto é visto como a última atividade de modelagem, antes da geração do código, ou seja, ele deve ser bem modelado, para que a etapa de construção ocorra sem alterações.”

(Pressman, 2011)

FASE PROJETO DE SOFTWARE

Projeto da Arquitetura do Software

- Componentes estruturais e relacionamentos

Projeto de Dados

- Estrutura de dados necessária

FASE PROJETO DE SOFTWARE

Projeto de Interfaces

- Comunicação interna, outros sistemas e usuários

Projeto de Componentes

- Procedimentos dos componentes arquiteturais

FASE PROJETO DE SOFTWARE



Fonte: Sommerville (2011).

CONCEITOS BÁSICOS DE PROJETO DE SOFTWARE

Abstração

- os elementos de projeto sejam representados por suas características essenciais, e os detalhes desnecessários sejam descartados.

Refinamento

- processo de elaboração parte do alto nível de abstração para níveis de abstração mais baixos

CONCEITOS BÁSICOS DE PROJETO DE SOFTWARE

Modularidade

- módulos/componentes que são integrados para corresponder aos requisitos levantados

Padrões

- soluções para problemas recorrentes em projetos de software

CONCEITOS BÁSICOS DE PROJETO DE SOFTWARE

Arquitetura

- estrutura ou organização dos módulos/componentes do programa e como eles interagem

Hierarquia de Controle

- estabelece os relacionamentos entre os componentes de software, seus níveis de abstração e refinamentos

CONCEITOS BÁSICOS DE PROJETO DE SOFTWARE

Encapsulamento

- define e impõe restrições de acesso tanto a detalhes procedurais em um módulo quanto em qualquer estrutura de dados local usada pelo módulo

Estrutura de Dados

- relacionamentos lógicos, como estão organizados os métodos de acesso, as associações e as alternativas de processamento de informações

CONCEITOS BÁSICOS DE PROJETO DE SOFTWARE

Procedimentos de Software

- Como a informação é processada, pontos de decisão, quais as sequências de evento, e que operações serão repetitivas

Ocultação de Informação

- Seu objetivo é propor uma maneira de decompor o problema, assim, obter módulos menores do software no desenvolvimento

“a qualidade de software é determinada pela qualidade dos processos que são usados durante a fase de desenvolvimento do software.”

Weber et al., 2001

QUALIDADE DO PROJETO

Qualidade do Processo

- qualidade dada ao produto final;
- avaliações no software acabado.

Qualidade do Produto

- modo como o software foi produzido ao longo do seu desenvolvimento
- como é feita a sua manutenção

QUALIDADE DO PROJETO

“Conformidade com as especificações” (Crosby, 1990).

O gerenciamento da qualidade deve ser feito desde o início do desenvolvimento, para tentar evitar defeitos e diminuir o retrabalho

QUALIDADE DO PROJETO

- O projeto deve implementar todas as especificações definidas nos requisitos.
- O projeto deve ser um guia legível para as próximas etapas do processo de software (implementação e teste).
- O projeto deve fornecer uma visão geral do software, como dados, possíveis funções e como ele deve se comportar.

PREMISSAS DA QUALIDADE

- Fazer acontecer
- Zero-defeito
- Qualidade = alto custo
- Padrões inibem a criatividade

MODELO DO PROJETO

O Modelo de Projeto possui quatro elementos que são considerados os principais e mais importantes:

- Arquitetura
- Dados
- Interfaces
- Componentes



DIMENSÕES DO MODELO DO PROJETO

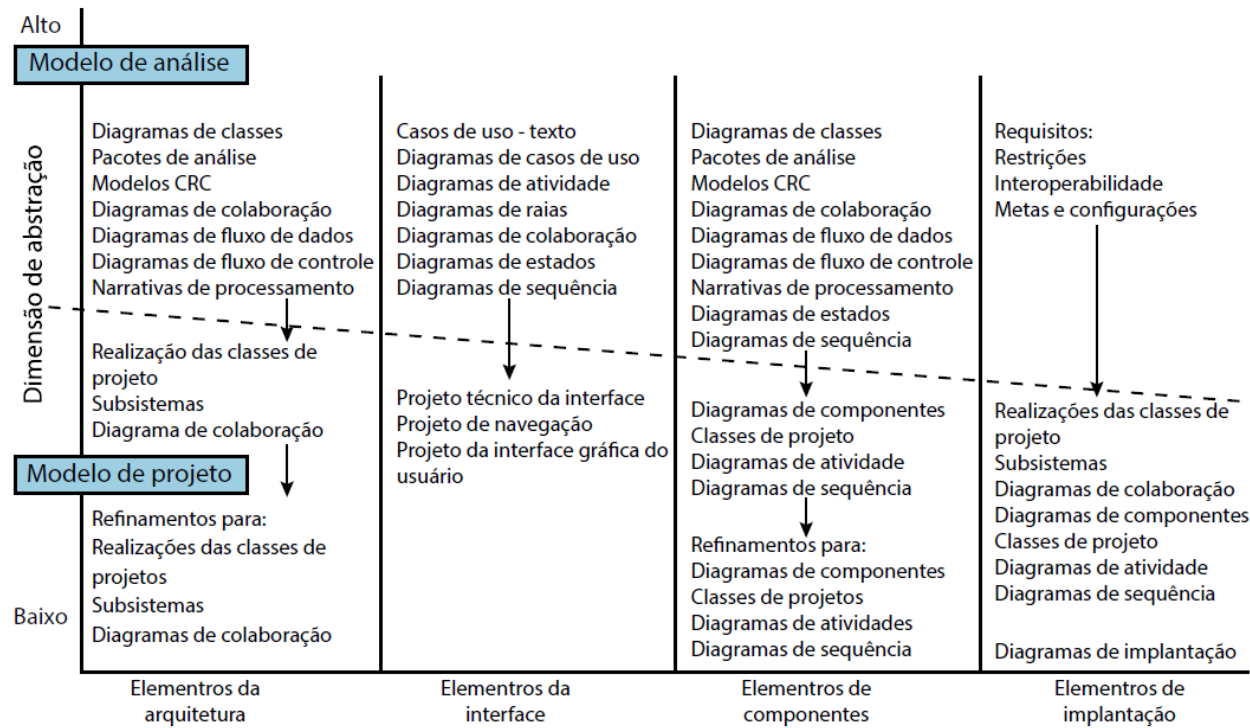
Processo

- mostra a evolução do projeto conforme as tarefas vão sendo executadas

Abstração

- mostra o nível de detalhamento de cada elemento do modelo de análise e que vão sendo transformados e refinados no modelo de projeto

DIMENSÕES DO MODELO DO PROJETO



Dimensão de Processo

MODELOS DO PROJETO

- Projeto da Arquitetura do Software
- Projeto de Componentes
- Projeto de Interface do Usuário
- Projeto de Dados

C4 MODEL

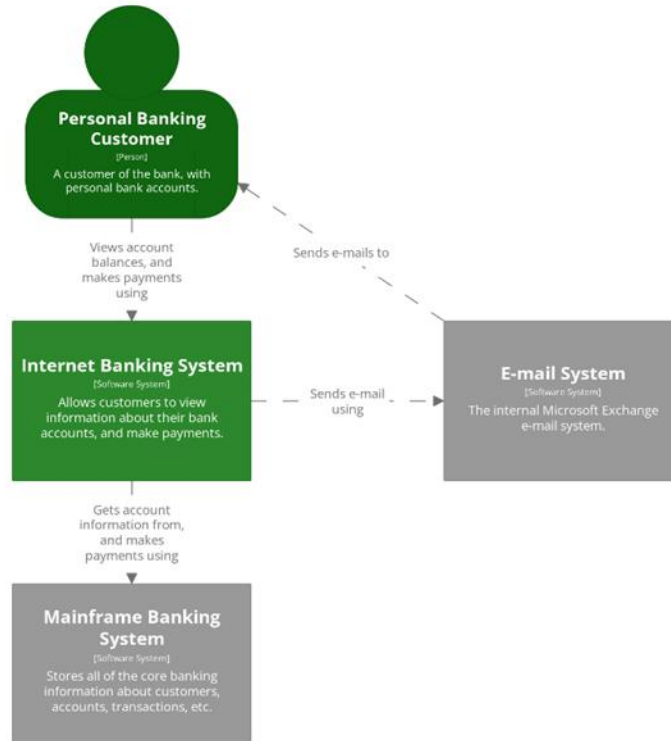
O modelo C4 é uma abordagem "abstração-primeiro" para diagramar arquitetura de software, com base em abstrações que refletem como arquitetos e desenvolvedores de software pensam e constroem software



DIAGRAMA DE CONTEXTO

- Fornece um ponto de partida, mostrando como o sistema de software em escopo se encaixa no mundo ao seu redor.
- Mostra as interações de forma macro, sem muitos detalhes, dando enfoque às comunicações e dependências entre sistemas e usuários que compõem e interagem com o software.
- É um diagrama que pode ser consumido por todas as “personas” do projeto, tanto técnico quanto de negócio, e que interagem direta ou indiretamente com o sistema.

DIAGRAMA DE CONTEXTO



Fonte: <https://c4model.com/>

DIAGRAMA DE CONTÊINER

- O segundo nível mostra o sistema com mais detalhes, descrevendo os seus containers e como eles se comunicam/interagem.
- A ideia é mostrar como o sistema é (ou será) construído de forma macro. Um container pode ser uma aplicação web, um banco de dados, um sistema de arquivos, entre outros.
- É um diagrama a ser consumido pela equipe técnica, que interage direta ou indiretamente com o sistema (profissionais de desenvolvimento, suporte etc).

DIAGRAMA DE CONTÊINER

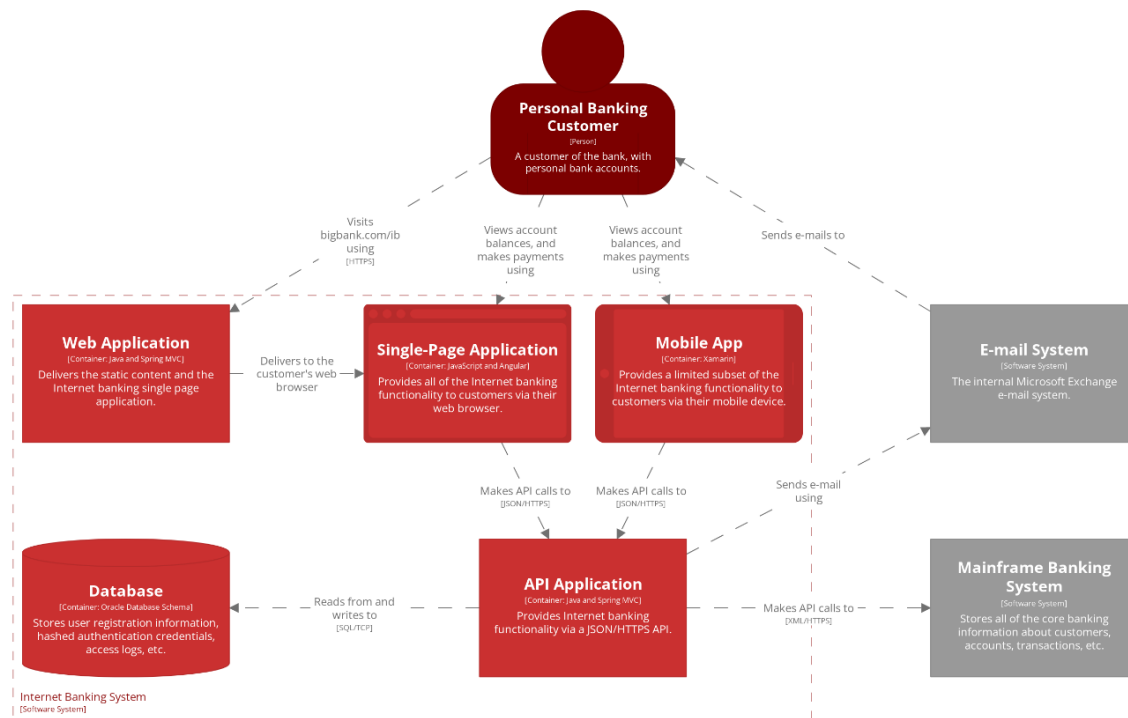


DIAGRAMA DE COMPONENTE

- Descrição das partes que compõem esses containers.
- Contêm informações sobre as interações, responsabilidades e tecnologias utilizadas de maneira mais detalhada que nos níveis anteriores.
- É um diagrama a ser consumido pela equipe técnica de desenvolvimento

DIAGRAMA DE COMPONENTE

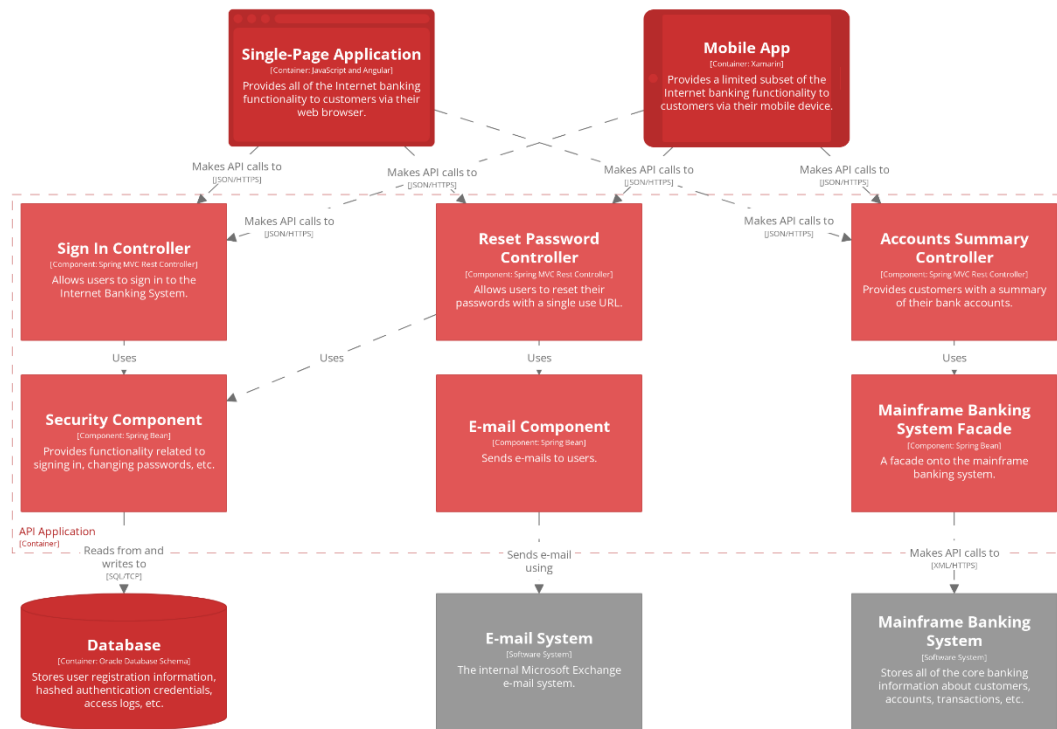
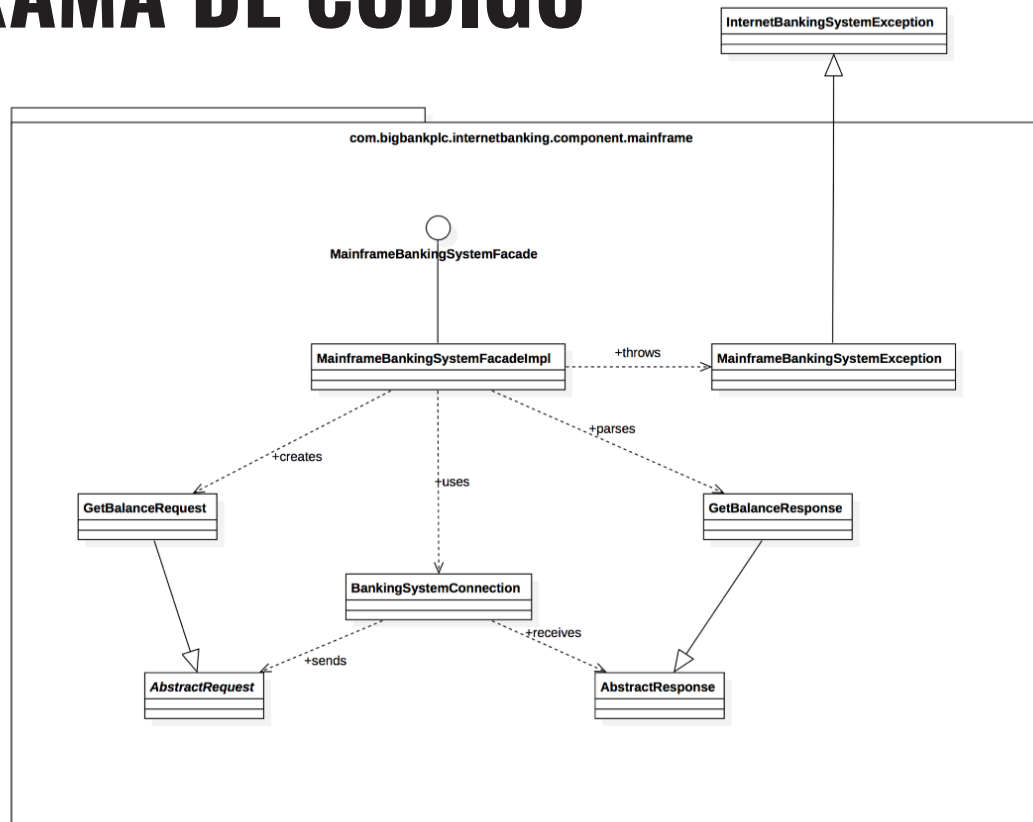


DIAGRAMA DE CÓDIGO

- Mostra, em nível de código, como cada componente é implementado e, para isso, usa o diagrama de classes do UML.
- Pode ser usado para ampliar um componente individual, mostrando como esse componente é implementado.

DIAGRAMA DE CÓDIGO



“O projeto é considerado o núcleo técnico da Engenharia de Software e, com isso ele pode ser aplicado em qualquer modelo de processo que seja adotado pela empresa. Por ser iniciado após a análise de requisitos, o projeto é visto como a última atividade de modelagem, antes da geração do código, ou seja, ele deve ser bem modelado, para que a etapa de construção ocorra sem alterações”

BONS ESTUDOS