Universidade Federal do Amazonas – **UFAM**
Institute of Computing - **IComp**
Usability and Software Engineering Group – **USES**

Arquitetura de software

Por que pensar nela e como registrá-la

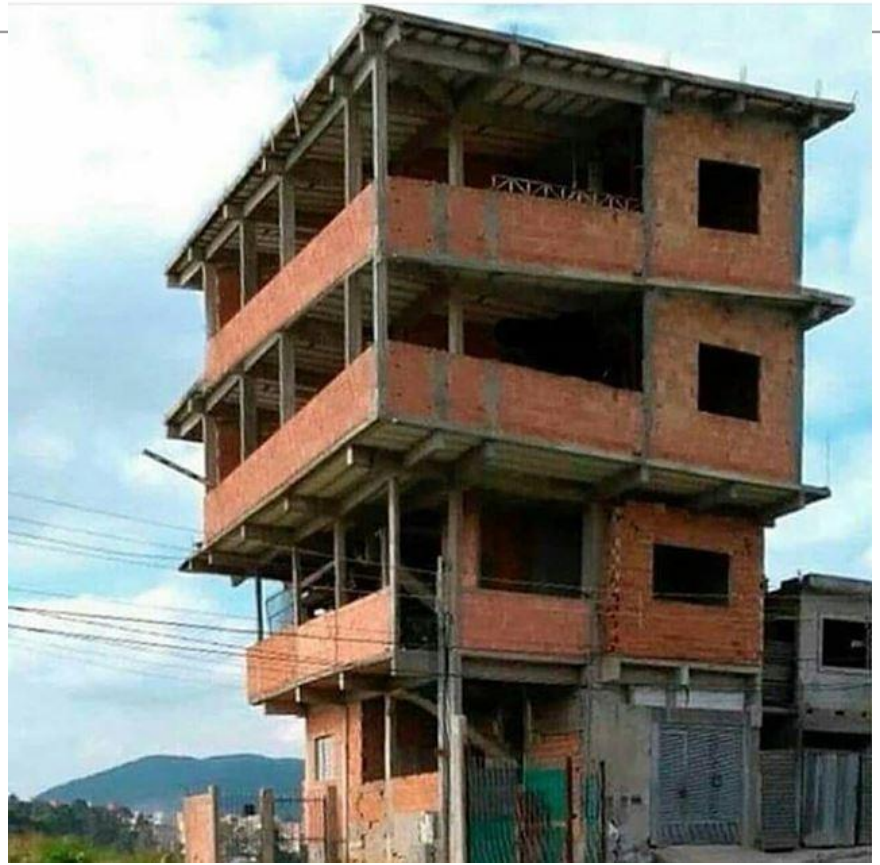
Slides feitos por Leonardo Barreto

Manaus, Brasil, 2023

Arquitetura

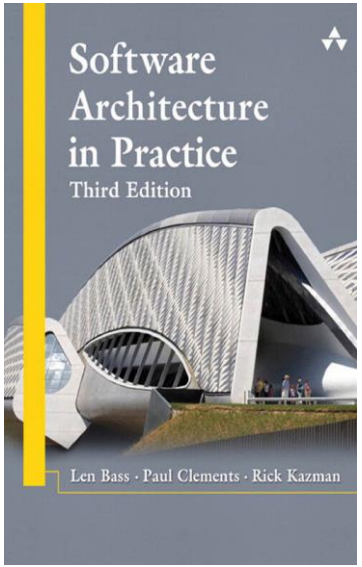
Arquitetura

Arquitetura



O que é arquitetura de software?

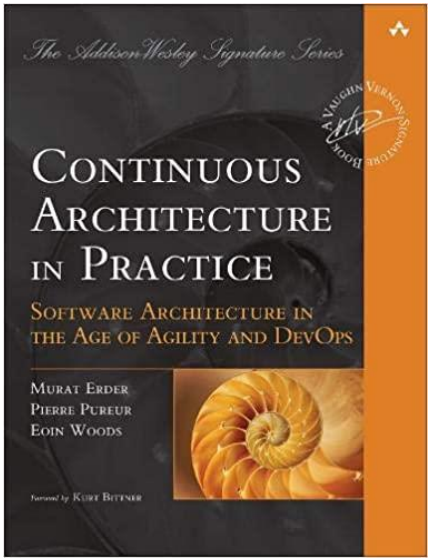
- Definição
  - **estrutura, ou estruturas, do sistema**,
  - abrange os **componentes de software**,
  - as **propriedades externamente visíveis** desses componentes
  - as **relações entre eles**.



Bass, L., P. Clements, and R. Kazman, Software Architecture in Practice, 2nd ed., Addison-Wesley, 2003

O que é arquitetura de software?

- Importância
  - **Qualidade**
    - **Escala**
    - **Segurança**
    - **Desempenho**
    - **Disponibilidade**
  - **Princípios sobre o sistema**
  - **Planejamento do futuro**



Erder M., Pureur P., Woods E., Continuous Architecture in Practice, 1st ed., Addison-Wesley, 2021

O papel do arquiteto

Entender o que o cliente quer para sua casa

Como traduzir e melhorar o que foi pensado em uma construção de uma casa

Repassar o conhecimento para quem vai construir a casa e cuidar do processo de construção

O papel do arquiteto de software

Entender o que o cliente quer para **o sistema (elicitar requisitos)**

Como traduzir e melhorar o que foi pensado em uma construção de uma casa **(analisar e projetar o sistema)**

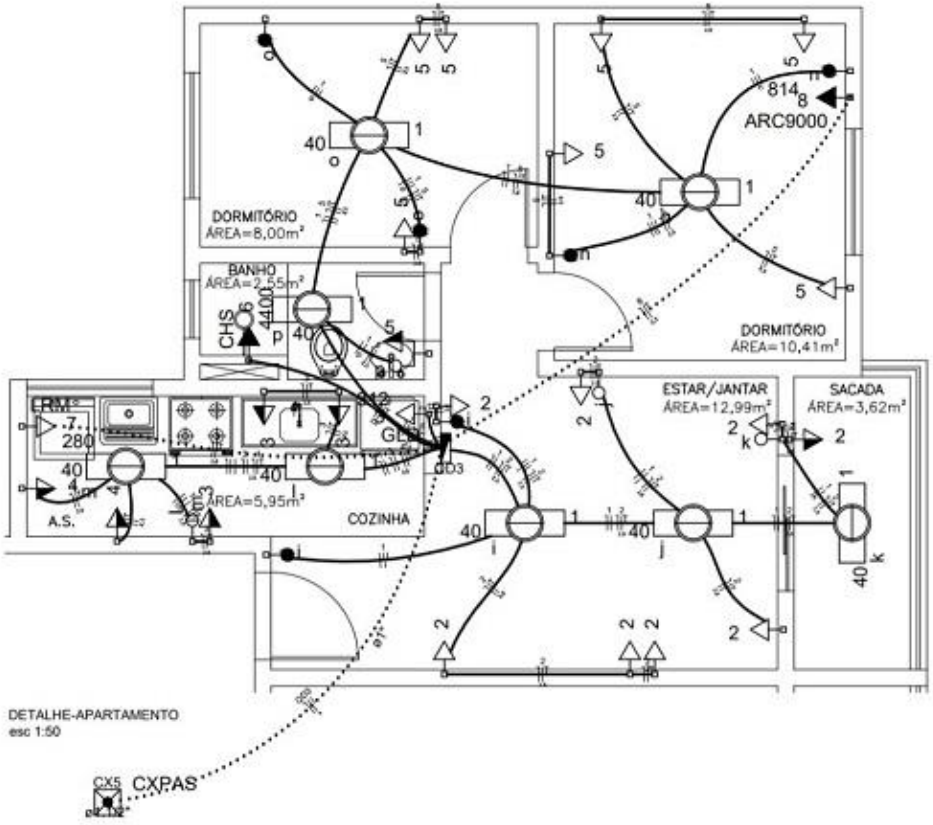
Repassar o conhecimento para quem vai construir a casa e cuidar do processo de construção **(desenvolver)**

Normalmente, o professional de TI faz as 3 coisas

Registrar a Arquitetura

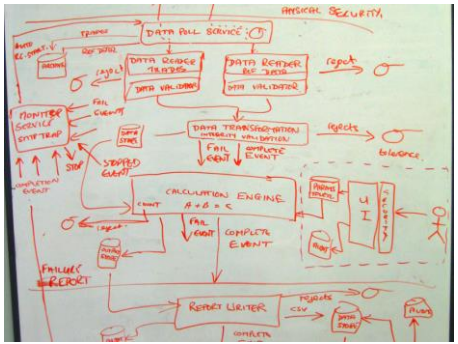
Como registrar a arquitetura de uma casa?

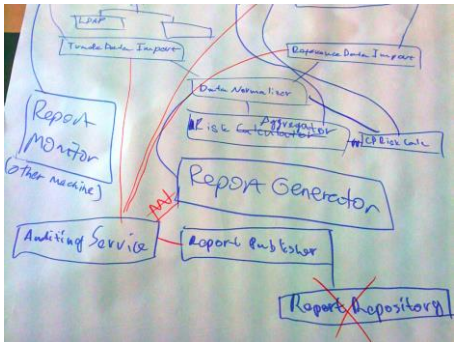
Como registrar a arquitetura de uma casa?

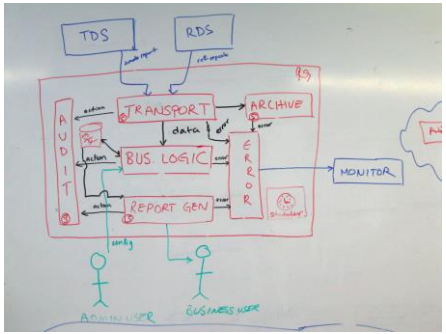


Como registrar a arquitetura de software?

- Não há padrões para desenhar a arquitetura de um software.

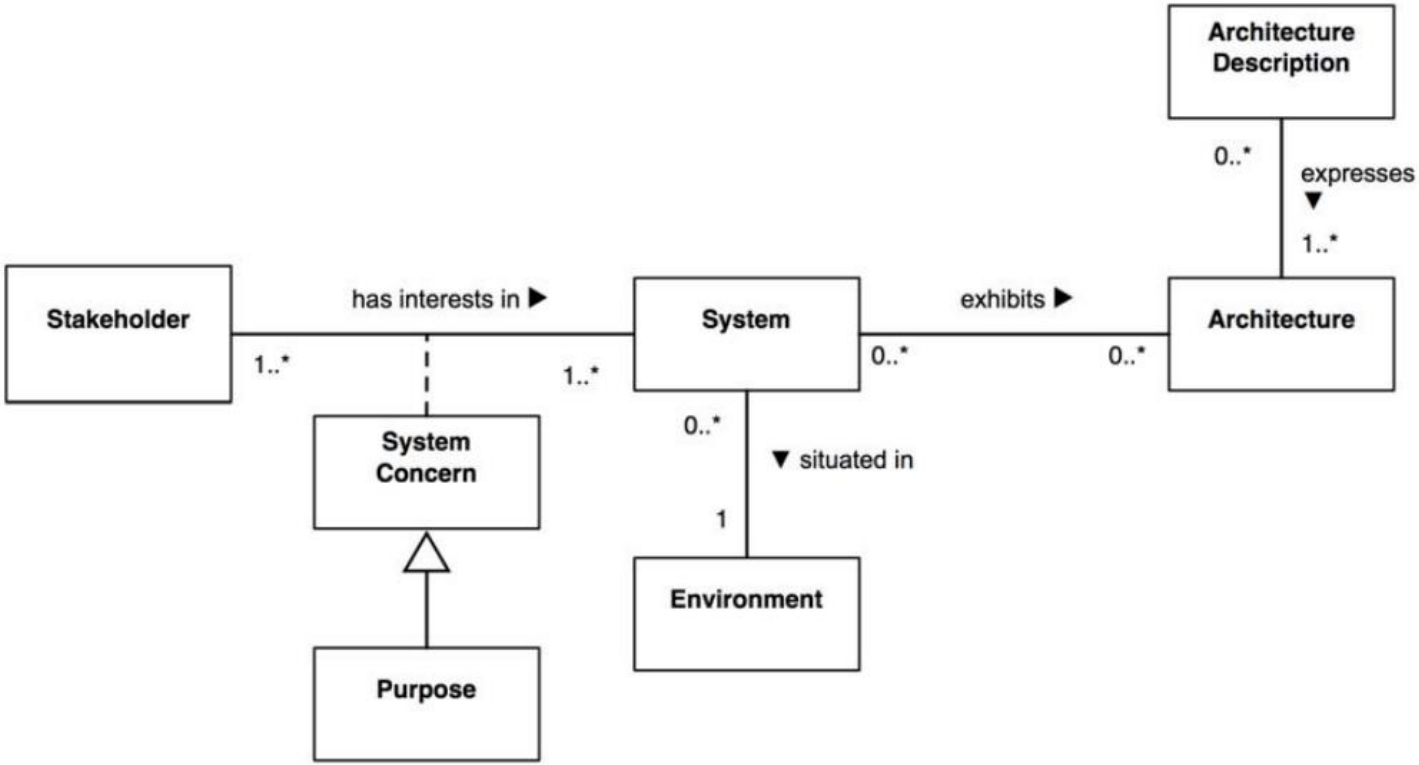






Imagens extraídas de www.c4model.com

ISO 42010:2011



"ISO/IEC/IEEE Systems and software engineering -- Architecture description," in ISO/IEC/IEEE 42010:2011(E) (Revision of ISO/IEC 42010:2007 and IEEE Std 1471-2000) , vol., no., pp.1-46, 1 Dec. 2011, doi: 10.1109/IEEESTD.2011.6129467.

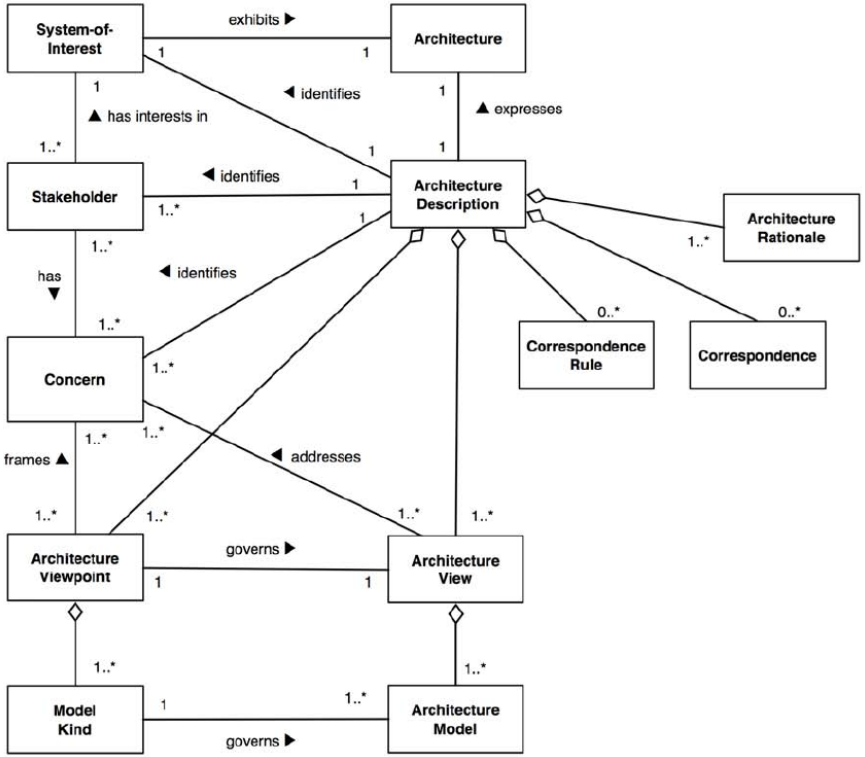
ISO 42010:2011

- Stakeholders (para quem eu quero mostrar algo do sistema)
  - usuários, operadores, donos, fornecedores, desenvolvedores, mantenedores, etc.
- Concern (aquilo que eu quero mostrar do sistema)
  - propósito do sistema
  - viabilidade da arquitetura escolhida para
    - alcançar o propósito
    - o sistema ser desenvolvido e implementado
  - riscos e impactos do sistema para os stakeholders
  - manutenção e evolução do sistema

"ISO/IEC/IEEE Systems and software engineering -- Architecture description," in ISO/IEC/IEEE 42010:2011(E) (Revision of ISO/IEC 42010:2007 and IEEE Std 1471-2000) , vol., no., pp.1-46, 1 Dec. 2011, doi: 10.1109/IEEESTD.2011.6129467.

ISO 42010:2011

- Architecture Description
  - visão da arquitetura
  - raciocínio
    - explicação das decisões importantes tomadas
  - modelos
    - notações, diagramas



"ISO/IEC/IEEE Systems and software engineering -- Architecture description," in ISO/IEC/IEEE 42010:2011(E) (Revision of ISO/IEC 42010:2007 and IEEE Std 1471-2000) , vol., no., pp.1-46, 1 Dec. 2011, doi: 10.1109/IEEESTD.2011.6129467.

Abordagens de descrição

Sistema de exemplo

- iFood (MVP)
- Requisitos funcionais
  - RF1 - O sistema deve ser capaz de intermediar compras entre os usuários e os restaurantes
  - RF2 - O sistema deve ser capaz de permitir o cadastro dos seus clientes
  - RF3 - O sistema deve listar os restaurantes de acordo com a sua localização inserida
  - RF4 - O sistema deve listar o cardápio do restaurante
  - RF5 - O sistema deve permitir que os restaurantes modifiquem seus cardápios
  - RF7 - O sistema deve identificar a disponibilidade do restaurante
  - RF21 - O usuário deve ser capaz de editar seus dados

Requisitos adaptados de https://github.com/Requisitos-2018-1-iFood/iFood/wiki/Prioriza%C3%A7%C3%A3o-de-requisitos---Moscow

Sistema de exemplo

- iFood (MVP)
- Requisitos funcionais
  - RF25O usuário deve ser capaz de visualizar o status do pedido em: Realizado, Não Realizado
  - RF27O sistema deve mostrar o “carrinho”, contendo a lista de comidas selecionadas e a opção de adicionar mais itens.
  - RF28Ao visualizar o carrinho, o sistema deve mostrar a opção de confirmação do pedido.
  - RF32Ao realizar o pedido, o sistema deve permitir ao usuário incluir adicionais ao seu pedido.
  - RF33Antes de adicionar ao carrinho, o sistema deve permitir a inclusão de observações ao pedido.
  - RF34O sistema deve permitir a alteração de quantidades dos itens pedidos no carrinho
  - RF37O sistema deve permitir ao restaurante incluir imagem e descrição dos pratos ofertados.
  - RF43O sistema deve permitir ao usuário o login.

Requisitos adaptados de https://github.com/Requisitos-2018-1-iFood/iFood/wiki/Prioriza%C3%A7%C3%A3o-de-requisitos---Moscow

C4 Model

- Criada por Simon Brown
- Inspirado no 4+1 e na UML
- 4 visões principais
  - **C**ontexto
  - **C**ontainers
  - **C**omponentes
  - **C**ódigo
- 3 visões complementares (opcionais)
  - Landscape, Dynamic, Deployment

Simon Brown, "The C4 model for visualising software architecture". Disponível em www.c4model.com

C4 Model - notação

- Caixas e setas (pode ter formas também)
- Todos os diagramas **devem ter**
  - um **título** e
  - uma **legenda**, explicando cada elemento
- Todos os elementos **devem ter**
  - um **tipo**
  - uma **descrição breve das suas responsabilidades**
  - cada container e componente **deve ter a sua tecnologia descrita**
- Todos os relacionamentos **devem**
  - **ser unidirecionais**
  - **ser rotulados de forma específica**
  - **conter o protocolo de comunicação, se for entre containers**

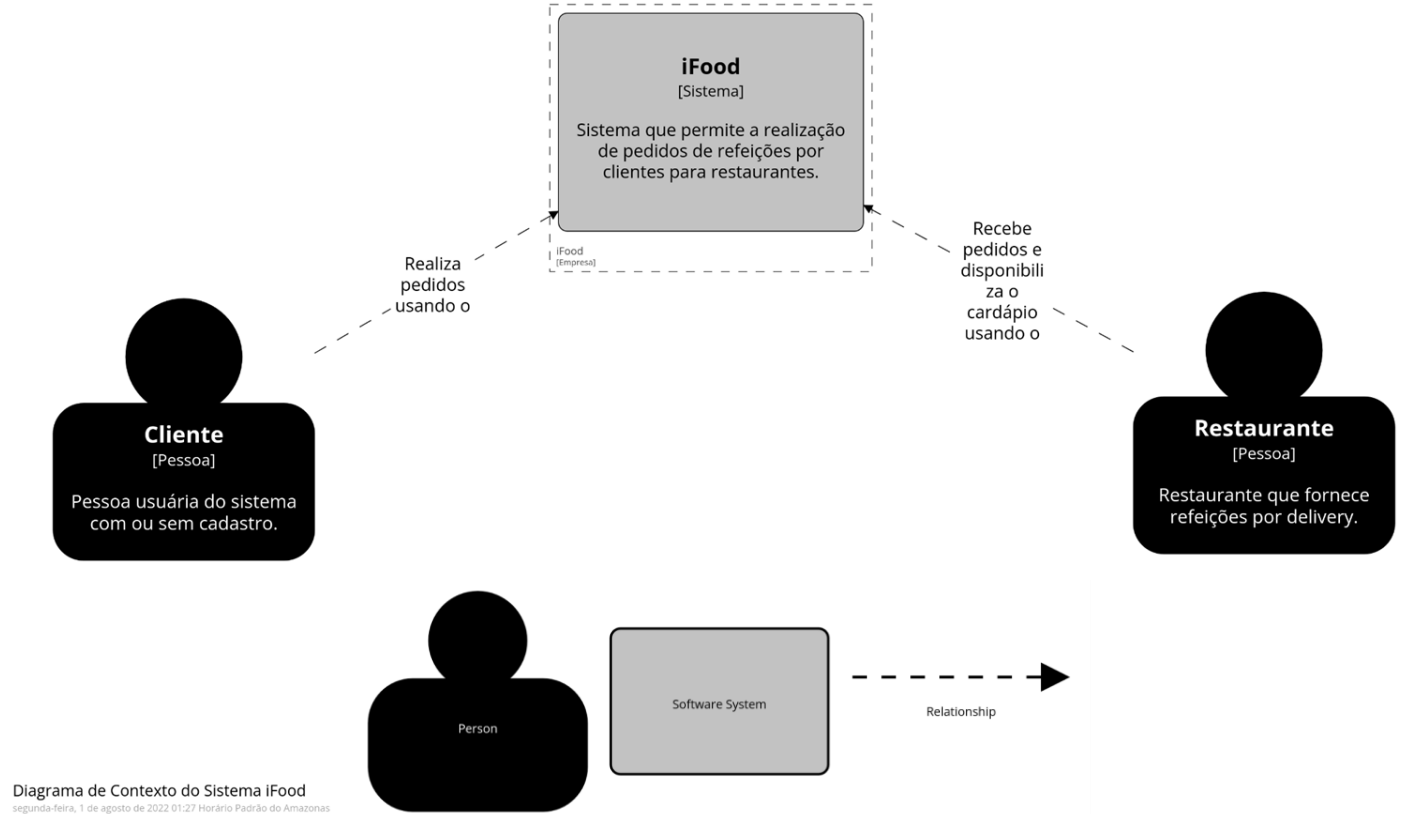
Simon Brown, "The C4 model for visualising software architecture". Disponível em www.c4model.com

C4 Model - Contexto

- **Stakeholders:** todo mundo da empresa ou fora dela
- **Concerns**: Atores e sistemas envolvidos no produto

Simon Brown, "The C4 model for visualising software architecture". Disponível em www.c4model.com

C4 Model - Contexto



C4 Model - Containers

- **Stakeholders:** pessoas técnicas da empresa
  - arquitetos, desenvolvedores, gerentes de projeto
- **Concern:** decisões tecnológicas, desenho do sistema e como as responsabilidades estão distribuídas.
  - Containers são as partes que integram o sistema

Simon Brown, "The C4 model for visualising software architecture". Disponível em www.c4model.com

C4 Model - Containers

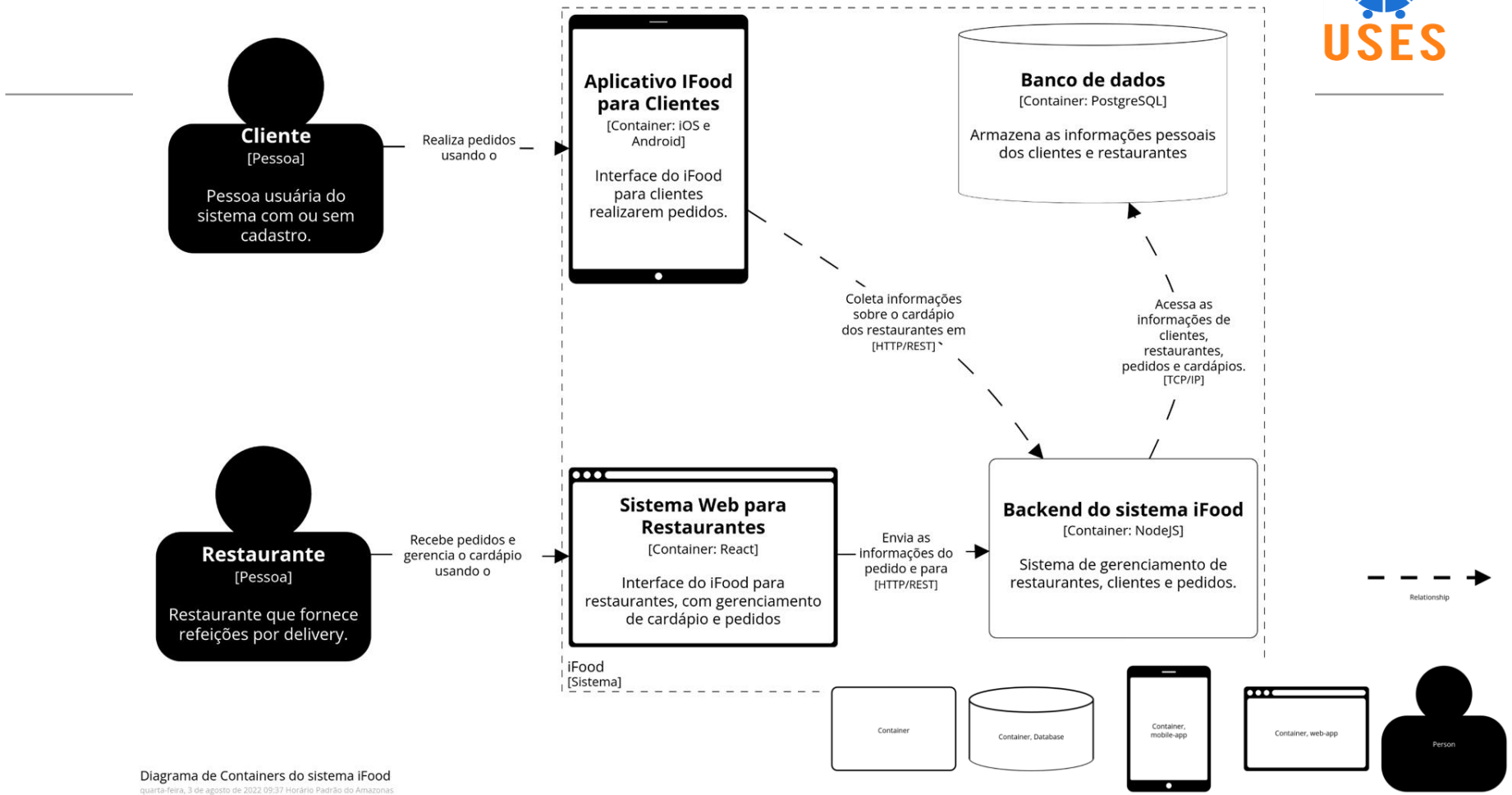


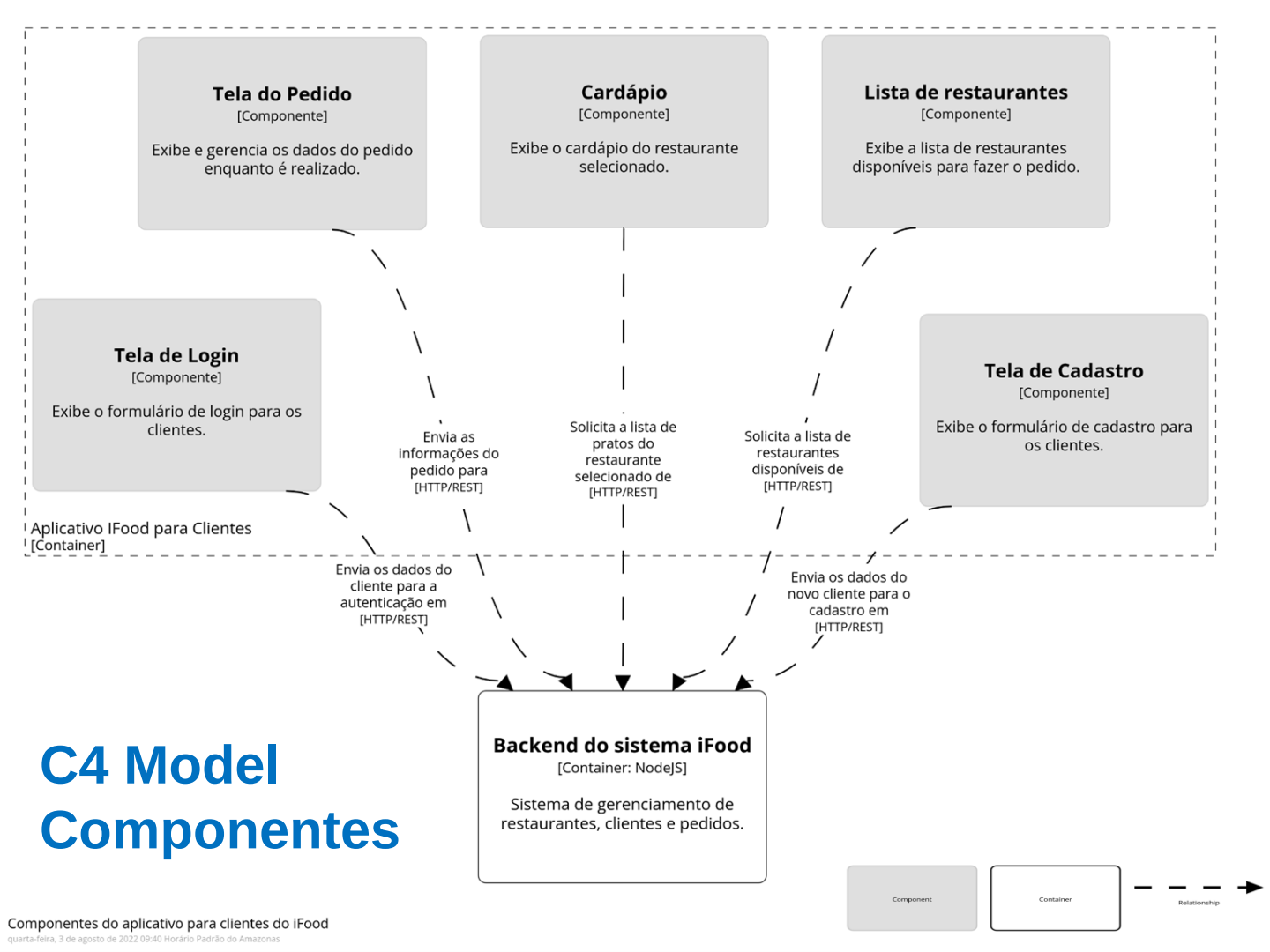
Diagrama de Containers do sistema iFood

quarta-feira, 3 de agosto de 2022 09:37 Horário Padrão do Amazonas

C4 Model - Componentes

- Partes integrantes dos containers
- **Stakeholders:** pessoas técnicas da empresa
  - arquitetos, desenvolvedores, gerentes de projeto
- **Concern:** organização do código-fonte em
  - Submodulos
  - Pacotes
  - Componentes

Simon Brown, "The C4 model for visualising software architecture". Disponível em www.c4model.com



C4 Model
Componentes

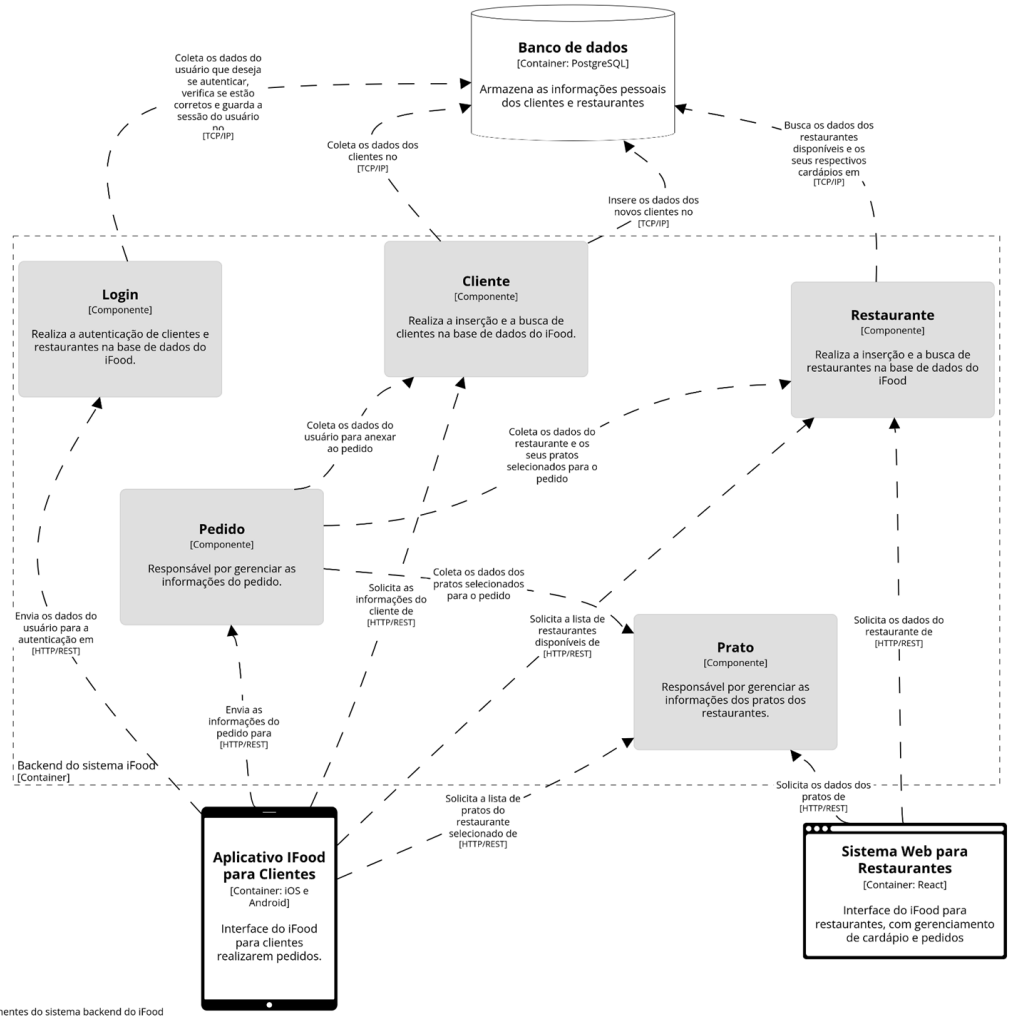
Componentes do aplicativo para clientes do iFood

quarta-feira, 3 de agosto de 2022 09:40 Horário Padrão do Amazonas

Backend do sistema iFood
[Container: NodeJS]

Sistema de gerenciamento de restaurantes, clientes e pedidos.

C4 Model
Componentes



Componentes do sistema backend do iFood

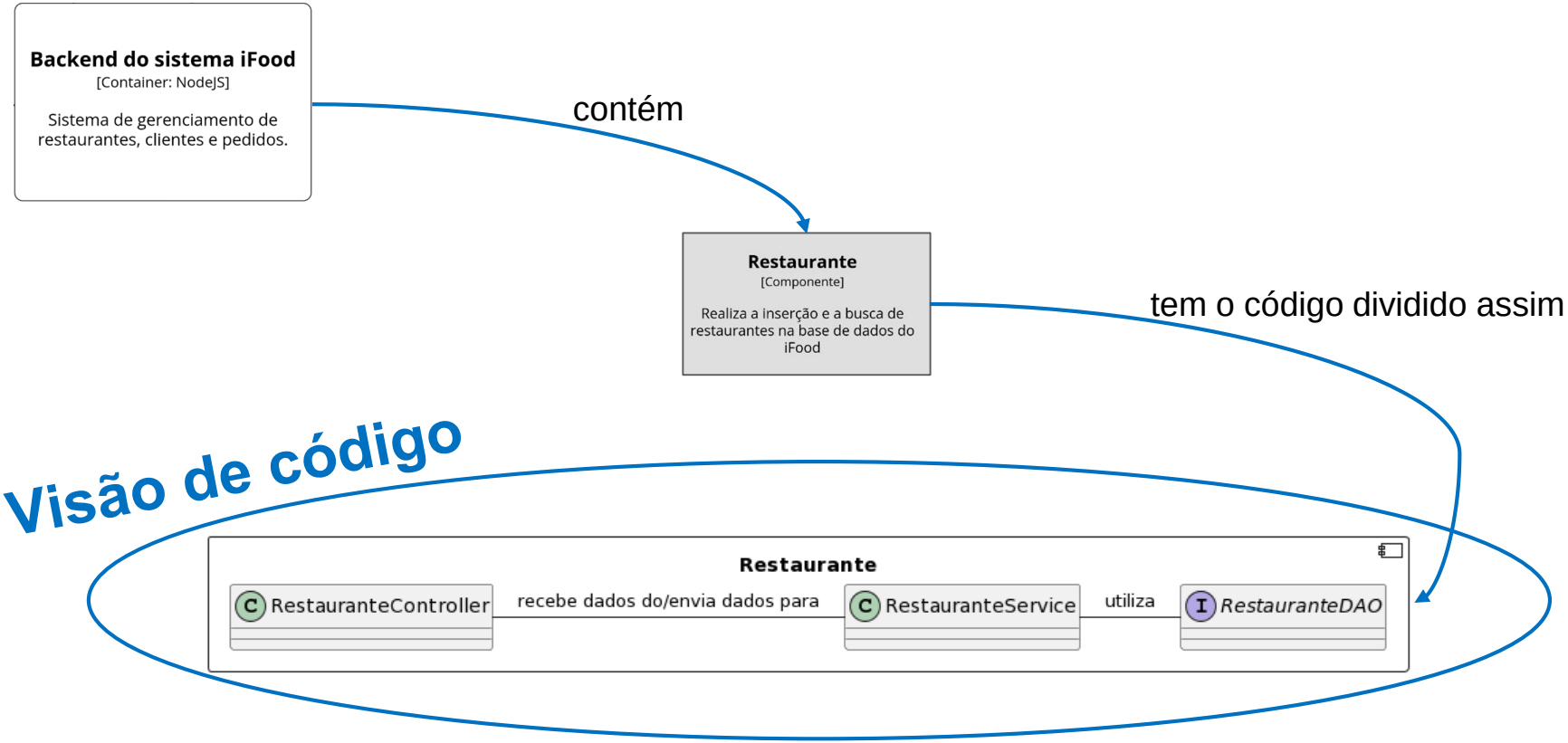
quarta-feira, 3 de agosto de 2022 09:40 Horário Padrão do Amazonas

C4 Model - Código

- **Stakeholders:** desenvolvedores
- **Concerns:** estrutura de código
- Como o sistema é implementado em código-fonte
- Diagramas UML (pacotes, componentes e classes)

Simon Brown, "The C4 model for visualising software architecture". Disponível em www.c4model.com

C4 Model - Código



Simon Brown, "The C4 model for visualising software architecture". Disponível em www.c4model.com

Extras

- Ferramentas
  - UML - Draw.io, PlantUML, Lucidchart
  - C4 - Structurizr, PlantUML C4
  - Papel e caneta

Simon Brown, "The C4 model for visualising software architecture". Disponível em www.c4model.com