

ES3 – Engenharia de Software III

Prof. Antonio Sergio Bernardo

PROJETO DE SOFTWARE

Exemplo Cantina (Cmaps)

Versão: 01/05/2024

LISTA DE FIGURAS

Figura 1	Nome1.....	x
Figura 2	Nome2.....	x
Figura 3	Nome3	x

LISTA DE QUADROS

Quadro 1	Nome1.....	x
Quadro 2	Nome2.....	x
Quadro 3	Nome3.....	x

SUMÁRIO

1 INTRODUÇÃO	X
2 FUNDAMENTOS DO PROJETO	X
2.1 CONTEXTO DE ATUAÇÃO	X
2.1.1 Canvas do Modelo de Negócio	X
2.1.2 Detalhamento das Atividades-Chave	X
2.2 ANÁLISE DO CONTEXTO	X
2.2.1 Problemas (P) e Oportunidades (O)	X
2.2.2 Soluções (S) e Iniciativas (I)	X
2.2.3 Conceitos e Dependências	X
3 REQUISITOS DE SOFTWARE	X
3.1 REQUISITOS FUNCIONAIS	X
3.1.1 Códigos, Nomes e Descrições	X
3.1.2 Diagrama de Casos de Uso	X
3.1.3 Descrição dos Casos de Uso	X
3.2 REQUISITOS NÃO FUNCIONAIS	X
4 DIAGRAMAS DE COMPORTAMENTO	X
4.1 SEQUÊNCIA	X
4.2 ATIVIDADES.....	X
4.3 ESTADOS.....	X
5 DIAGRAMAS ESTRUTURAIS	X
5.1 CLASSES	X
5.2 PACOTES	X
5.3 COMPONENTES	X
6 MODELAGEM DE DADOS	X
6.1 MODELO CONCEITUAL.....	X
6.2 MODELO LÓGICO	X
7 TECNOLOGIAS UTILIZADAS E APIS	X
8 TELAS DO PROTÓTIPO	X
9 RELATÓRIOS EMITIDOS	X
10 DOCUMENTOS GERADOS	X

1 INTRODUÇÃO

Conforme orientações do Template.

2 FUNDAMENTOS DO PROJETO

2.1 CONTEXTO DE ATUAÇÃO

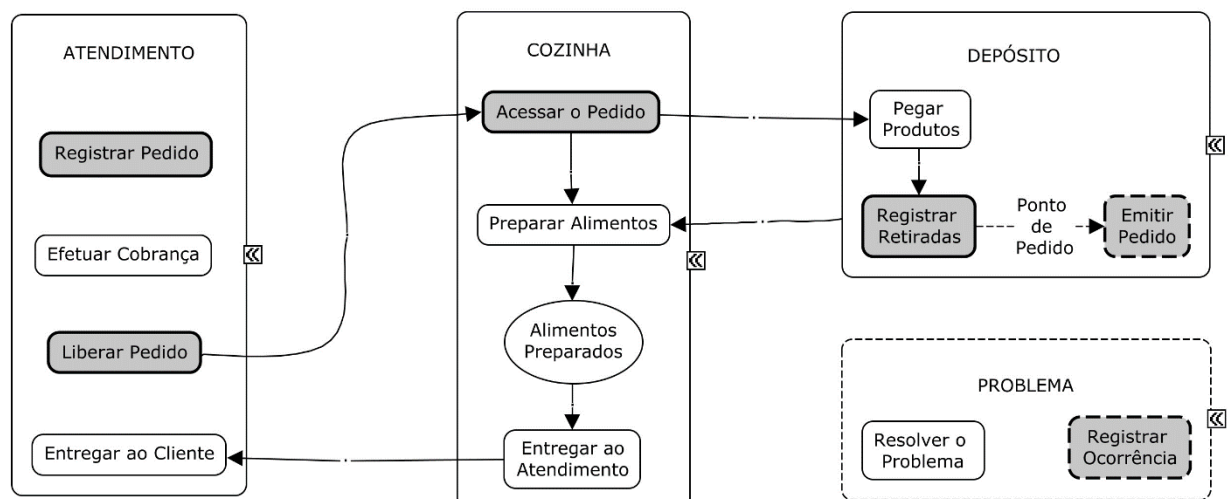
2.1.1 Canvas do Modelo de Negócio – Quadro 1

Canvas Word - Modelo de Negócio: Cantina

Parcerias Principais	Atividades-Chave	Propostas de Valor	Relacionamento	Segmentos de Clientes
Atacadista Assai Serviços de Limpeza Limpabem	ATENDIMENTO COZINHA DEPÓSITO	Comida Caseira Porções Variadas Chopp Cervejas Refrigerantes Sucos	Horário: 8 às 23 h Presencial Encomendas Site Institucional Loja Virtual (E-Commerce)	Indivíduos Casais Famílias Aniversários Grupos de Trabalho
	Recursos Principais	Cartões de Crédito Cartões de Débito Vales de Refeição	Canais	
	Ambiente Aconchegante Area Interna 30 Pessoas Area Externa 50 Pessoas Rampa de Acesso	Cartazes WhatsApp Telefone (15) 3333-4444 cantinal@hotmail.com WebSites		
Estrutura de Custos			Fontes de Receita	
Compras de Insumos, Serviços de limpeza, Pagamento de Funcionários, Impostos			Recebimento de Clientes	

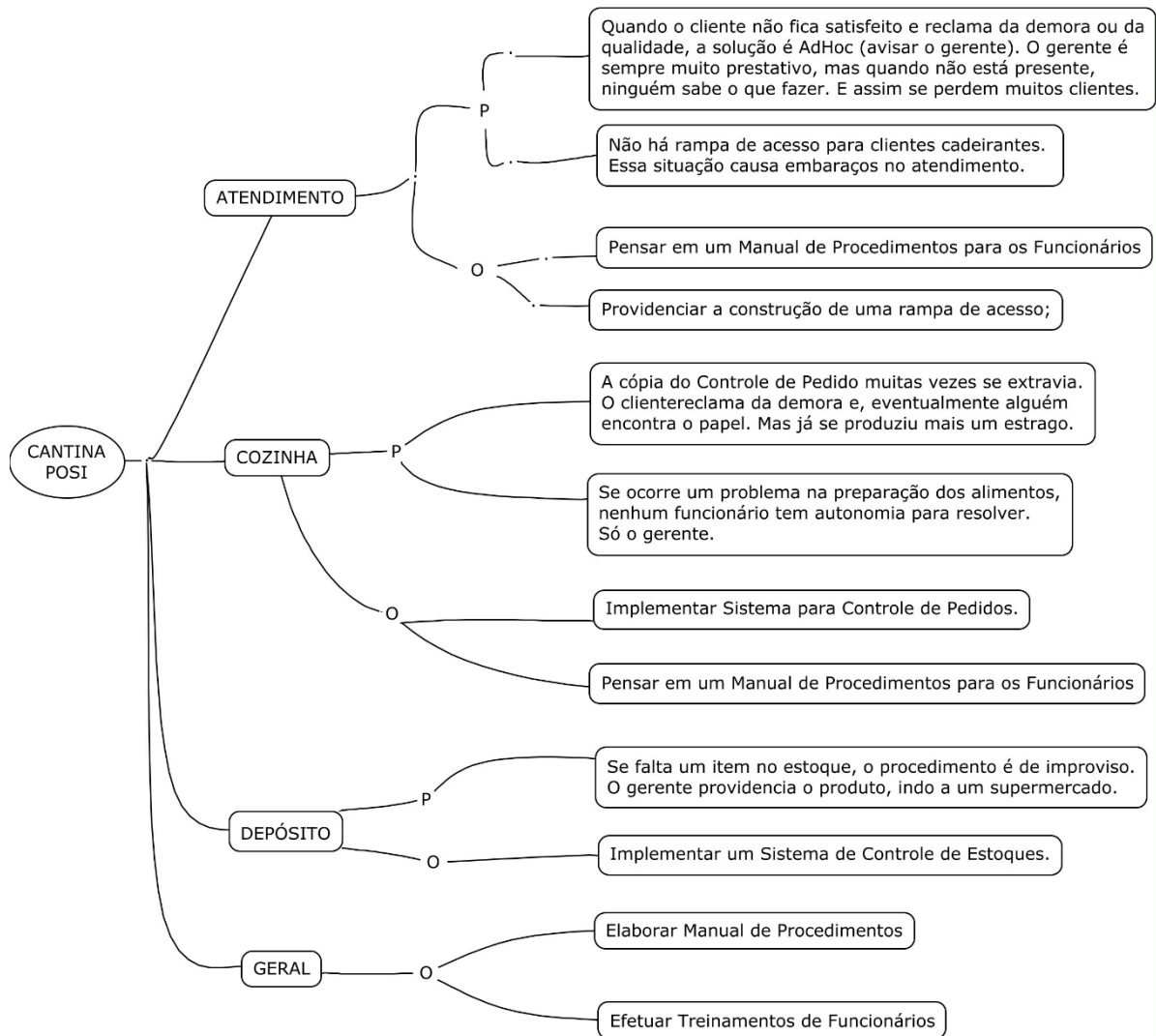
2.1.2 Detalhamento das Atividades-Chave (opcional) - Figura 1

Processo Cantina



2.2 ANÁLISE DO CONTEXTO (Histórias de Usuários)

2.2.1 Problemas (P) e Oportunidades (O) - Figura 2



2.2.2 Soluções (S) e Iniciativas (I)

Figura 3

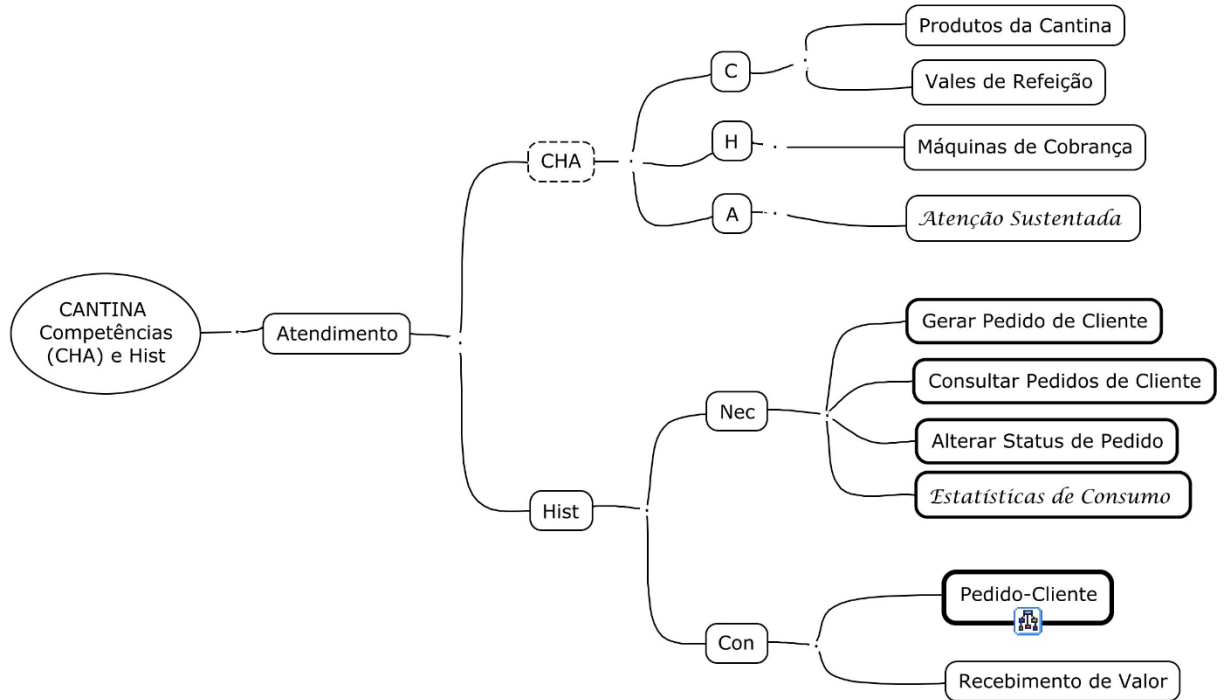


Figura 4

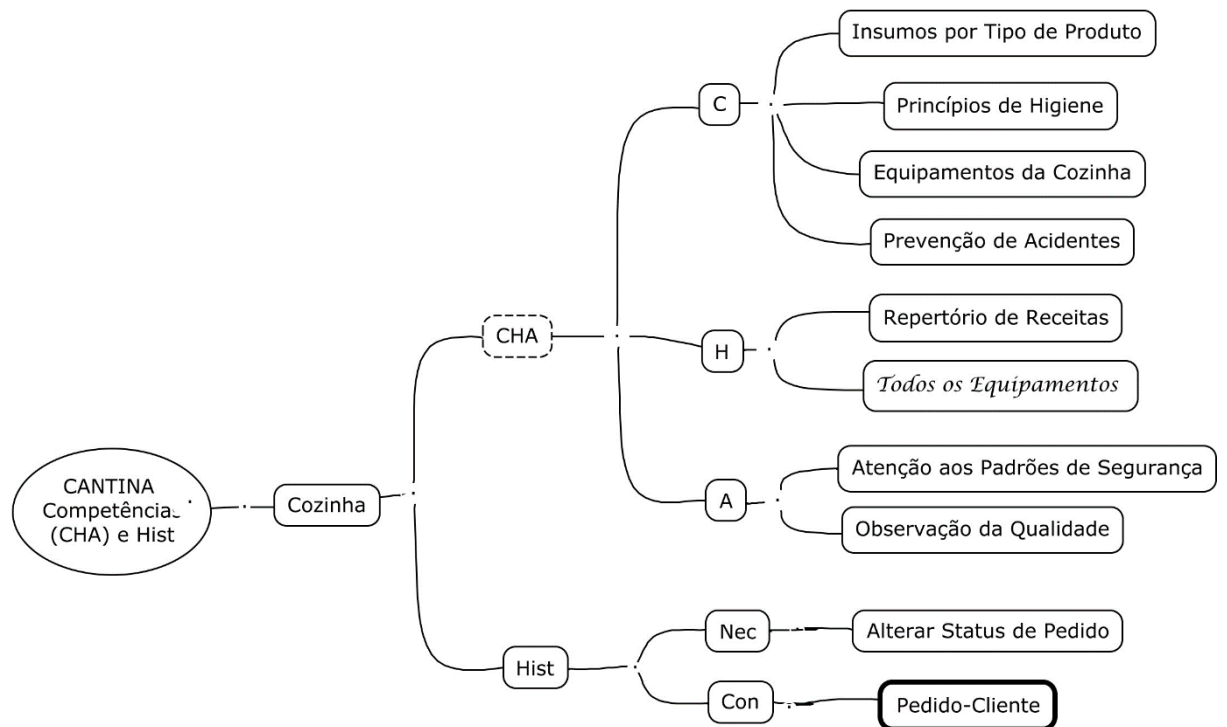
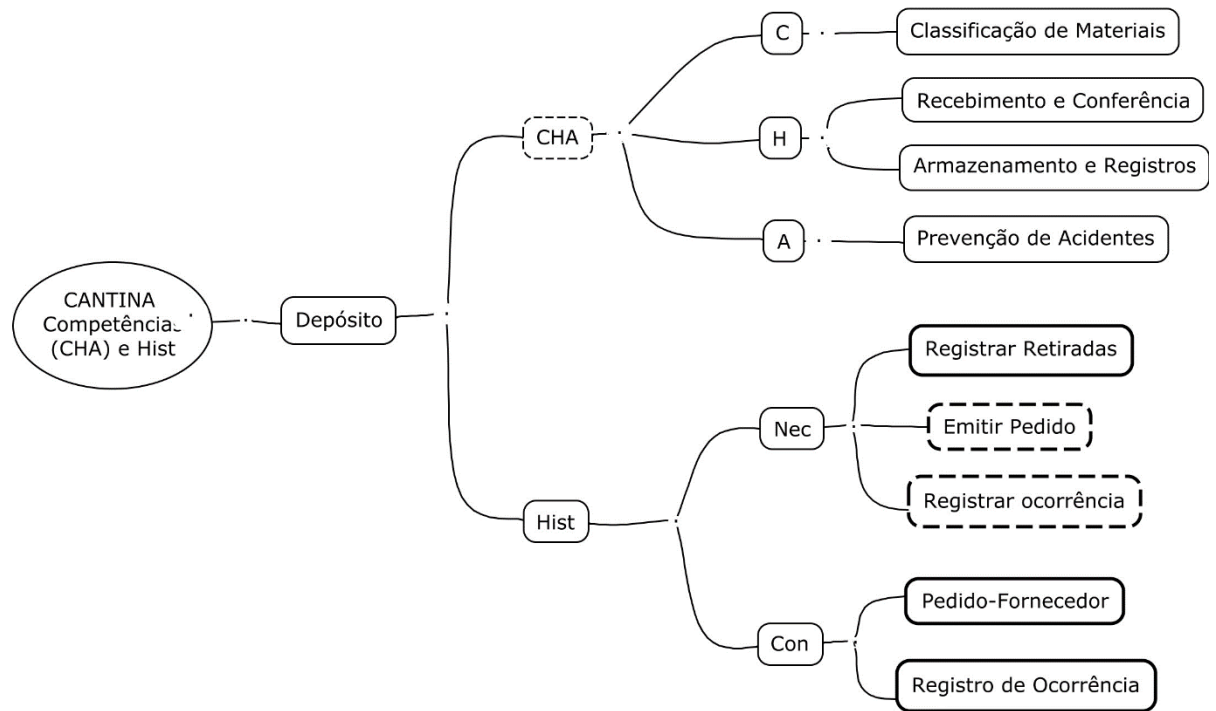
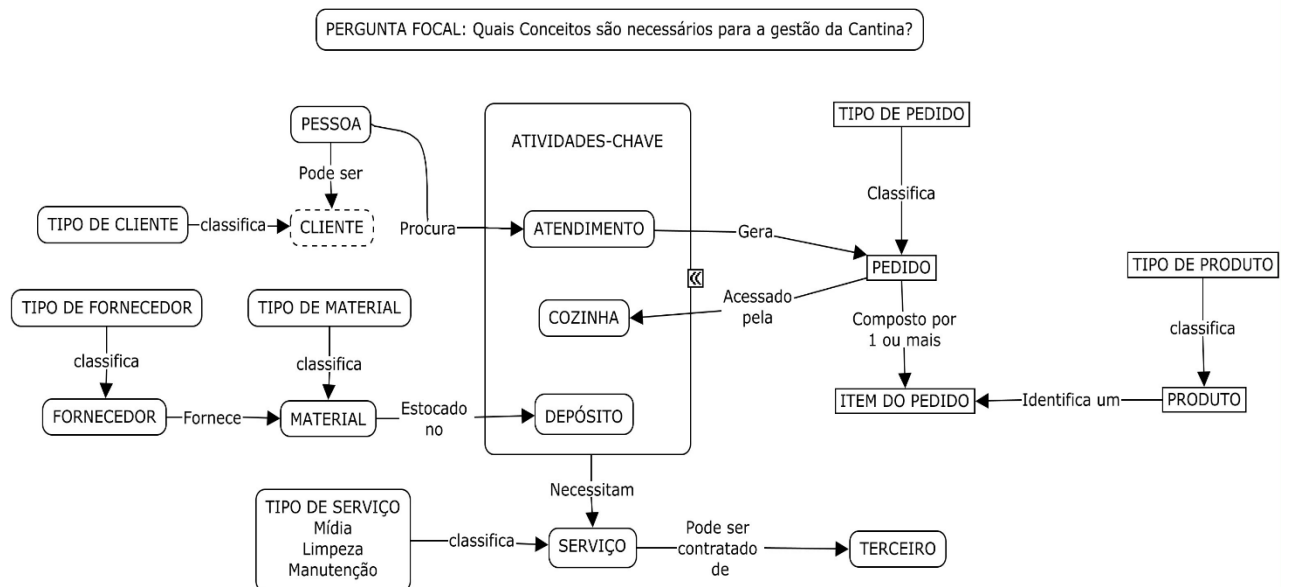


Figura 5



2.2.3 Conceitos e Dependências - Figura 6

CMAP - CONCEITOS DE NEGÓCIO



3 REQUISITOS DE SOFTWARE

3.1 REQUISITOS FUNCIONAIS

3.1.1 códigos, nomes e Descrições

RF1 – Gerar Pedido de Cliente

Início do Ciclo de Vida de Pedido do Cliente, efetuado na Recepção da Cantina.

RF2 – Consultar Pedidos de Cliente

Refere-se à necessidade que os funcionários têm de verificar a situação de um Pedido, seja para informar ao cliente, seja para efetuar os atendimentos.

RF3 – Alterar Status de Pedido do Cliente

Início do Ciclo de Vida de Pedido do Cliente, efetuado na Recepção da Cantina.

RF4 – Incluir Registro de Ocorrência

Requisito para manter um Histórico de Problemas, como arte importante do Monitoramento do Processo, que será útil nos Diagnósticos para Melhorias.

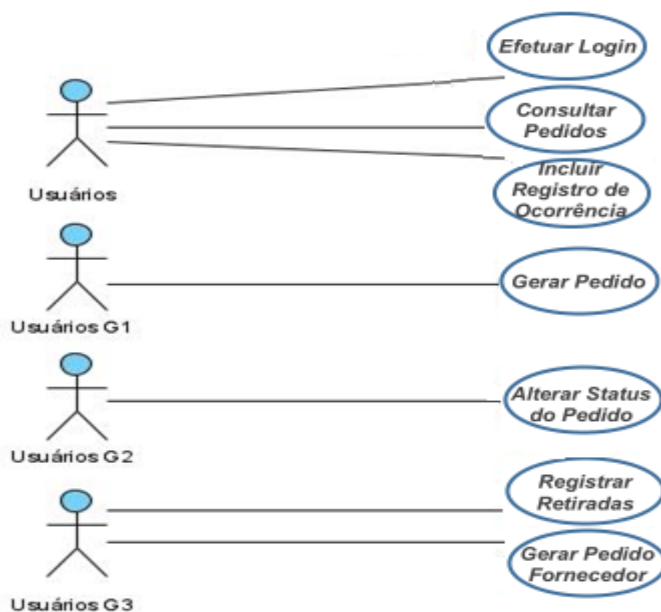
RF5 – Registrar Retirada de Insumos

Requisito para atender o Controle de Estoque, ligado ao Ponto de Pedido dos Insumos.

RF6 – Gerar Pedido para Fornecedor

Requisito para atender situações de Ponto de Pedido, decorrente da Retirada de Insumos para atendimento dos Pedidos.

3.1.2 DIAGRAMA DE CASOS DE USO (Figura 1)



3.1.3 DESCRIÇÃO DOS CASOS DE USO

Quadro 5. Caso de uso – Gerar Pedido de Cliente

Caso de Uso	RF1: GERAR PEDIDO	
Tarefa Executante	Registrar Pedido, na fase de Recepção	
Ator Principal	Usuário do Grupo 1	
Ator Secundário		
Pré-Condição	Login efetuado ok com apresentação do Menu do Sistema	
Pós-Condição		
Ações do Ator		Ações do Sistema
1 - Usuário clica no Menu a opção Gerar Pedido		
		2 - Apresenta a Tela correspondente à Função selecionada pelo Usuário
3 - Usuário informa Itens Solicitados, Confere, e clica Gravar		
		4 – Efetua gravação do Pedido e Retorna ao Menu

3.2 REQUISITOS NÃO FUNCIONAIS

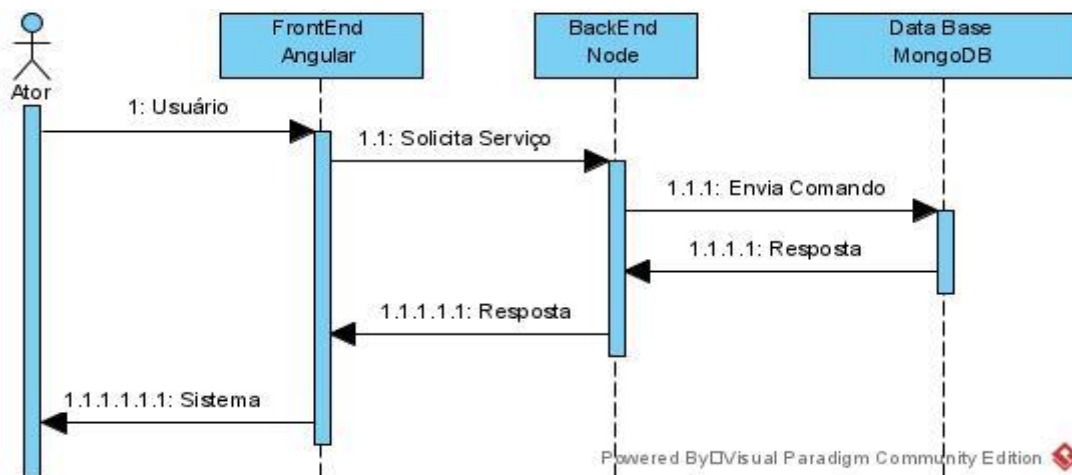
Quadro 6. Requisitos Não Funcionais

Código	Descrição do Requisito
RNF1	Usuários devem ter identificação única com Nome e Senha
RNF2	Deve haver um Administrador do Sistema para gerenciar os acessos
RNF3	Todo usuário deve ter a condição de alterar a sua própria senha
RNF4	Menus de operação do sistema diferenciados por Grupo de Usuários
RNF5	Permissões de acesso devem ter opção para Leitura ou Gravação
RNF6	As opções de Menu devem ter seus conteúdos registrados em tabelas
RNF7	O Sistema deve ser operacional em Desktop e em Dispositivos Móveis
RNF8	Produtos e Insumos devem ser controlados com Código de Barras
RNF9	O Sistema e a Base de Dados devem estar localizados em Nuvem
RNF10	A Cantina deve poder operar sem Sistema em caso de Contingência

4 DIAGRAMAS DE COMPORTAMENTO

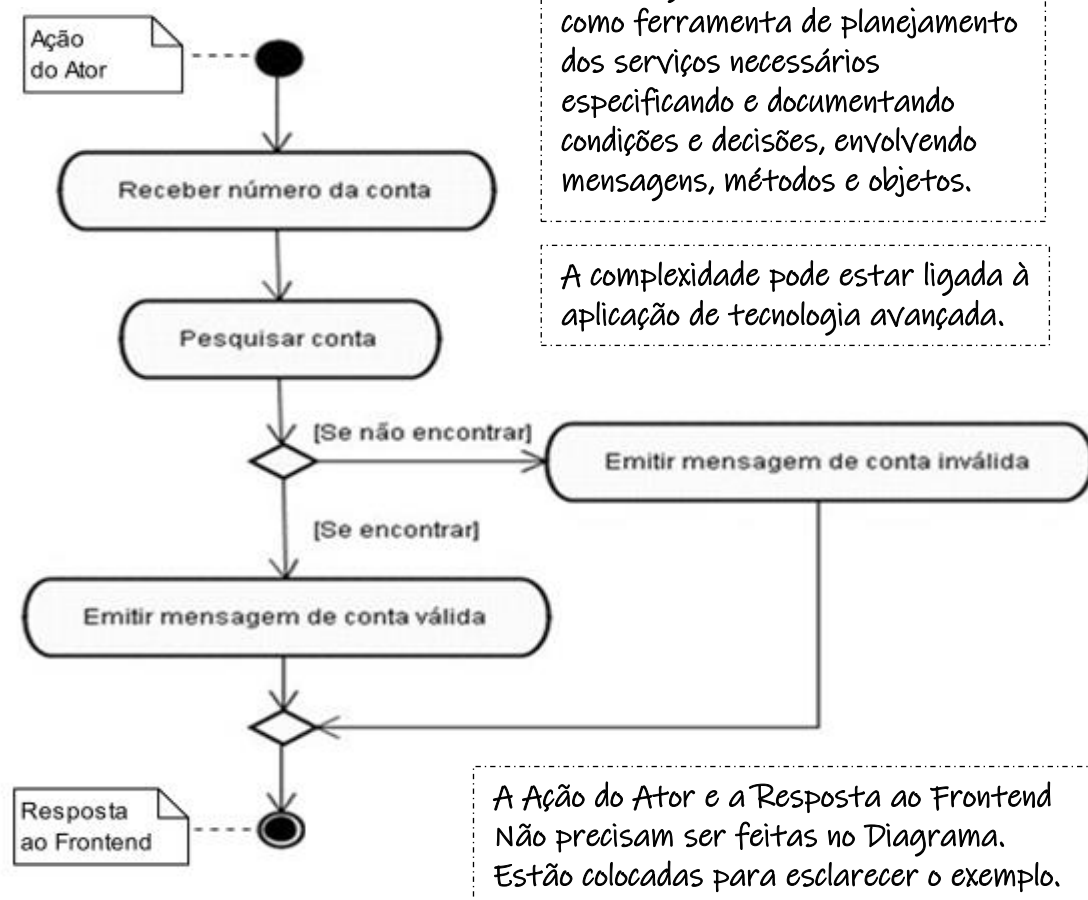
4.1 Diagrama de Sequência

Figura 2. Projeto em 3 camadas (MVC – Model, View, Controller).



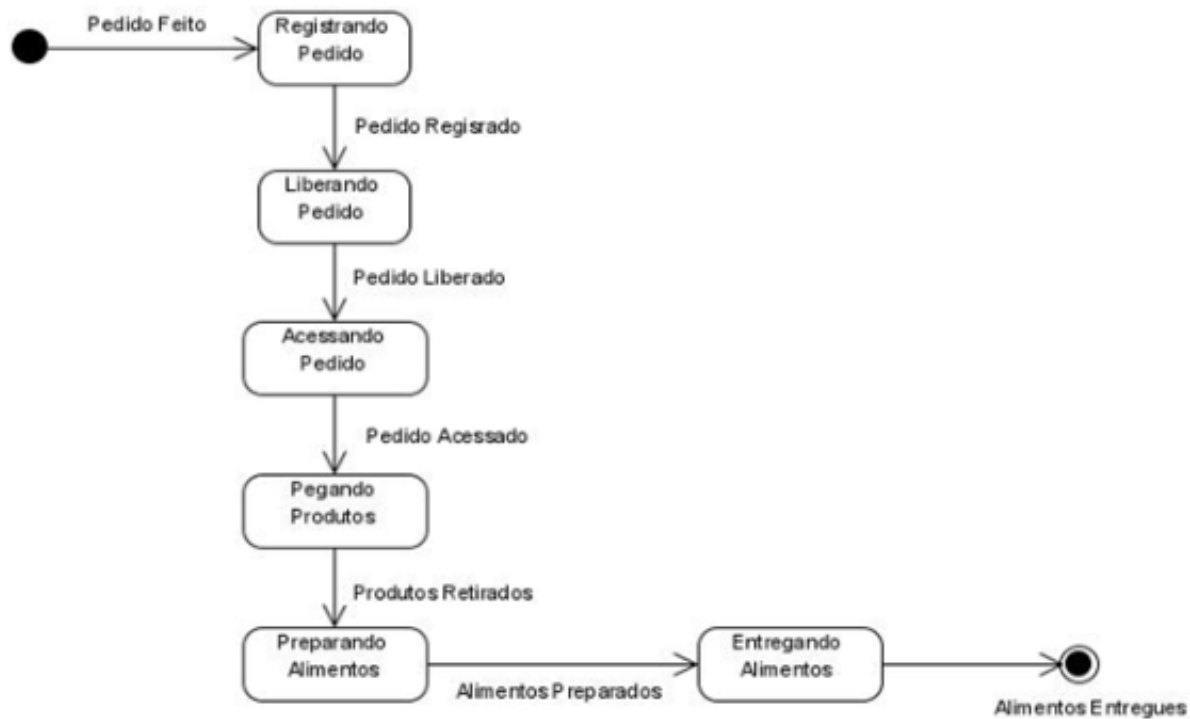
4.2 Diagrama (s) de Atividades →

Figura 3. RFn.Nome.CAU.n



4.3 Diagrama (s) de Estados

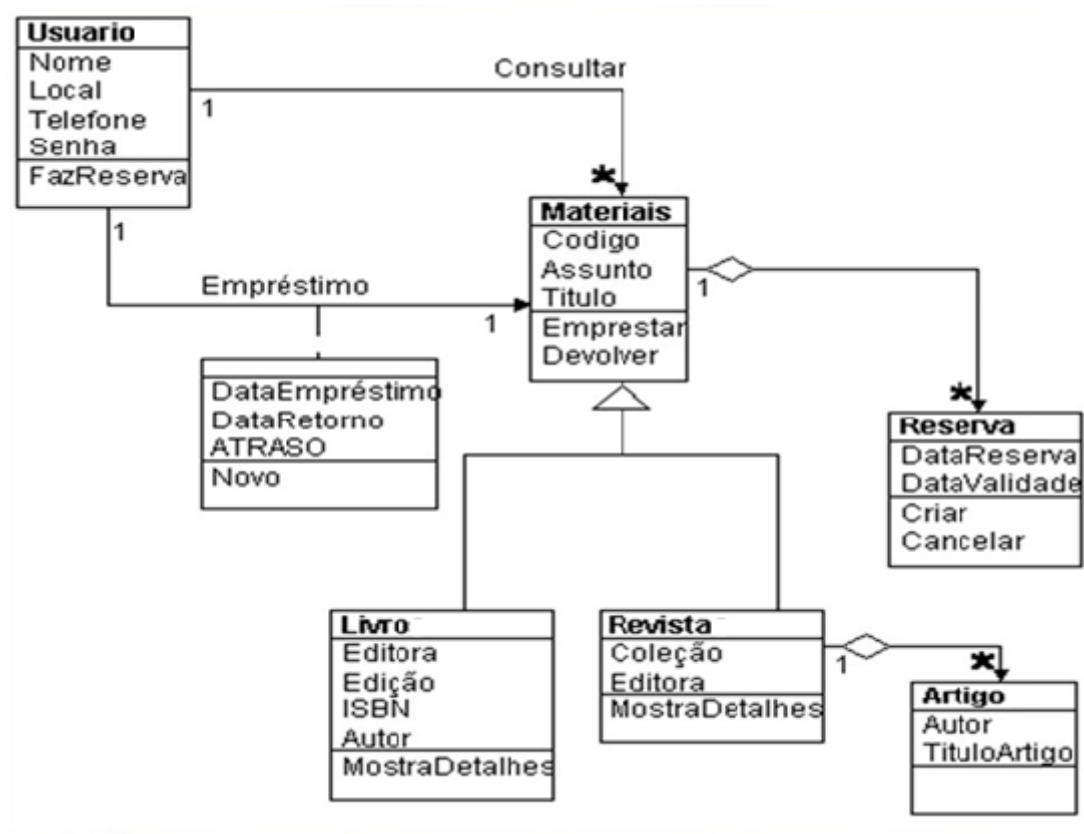
Figura 4. Ciclo de Vida do Pedido



5 DIAGRAMAS ESTRUTURAIS

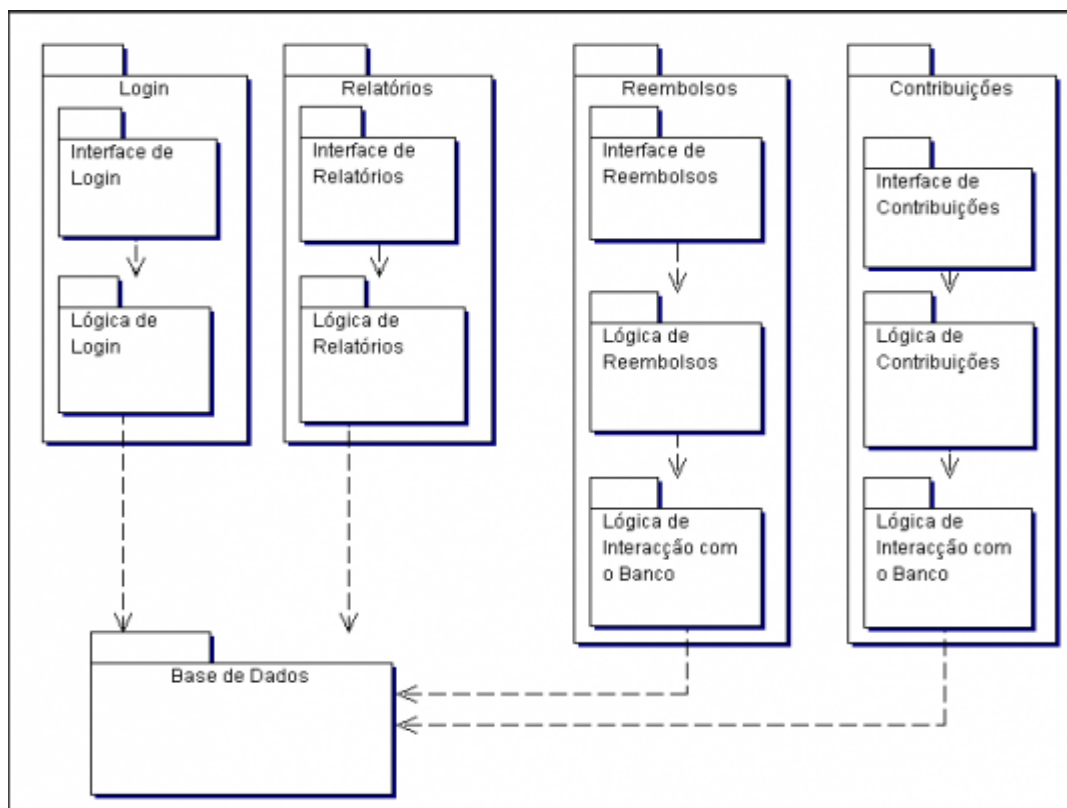
5.1 Diagrama de Classes

Figura 5. Diagrama de Classes (Exemplo)



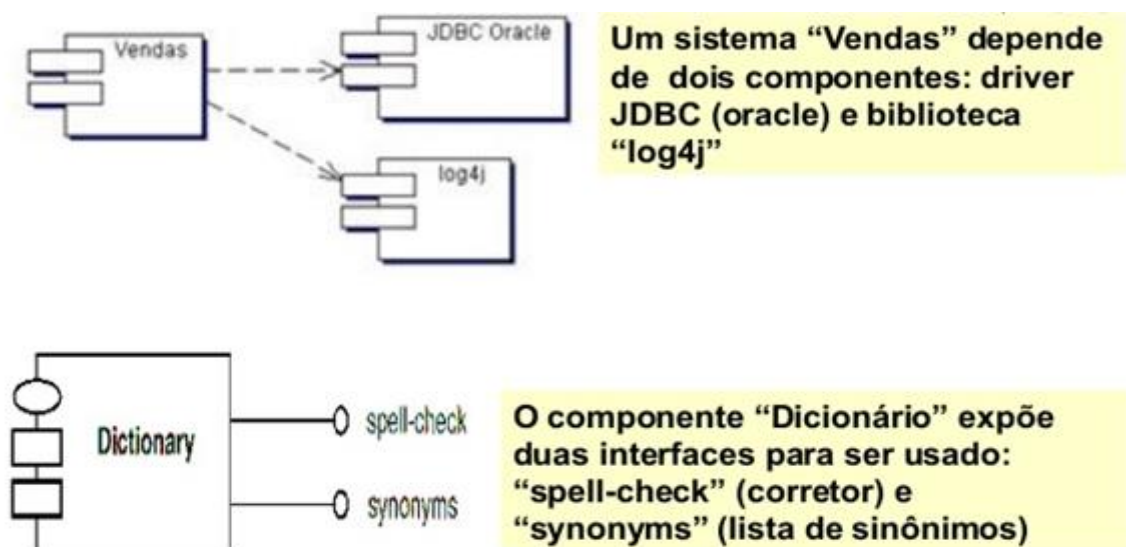
5.2 Diagrama de Pacotes

Figura 6. Diagrama de Pacotes (Exemplo)



5.3 Diagrama de Componentes

Figura 7. Diagrama de Componentes (Exemplo)



6 MODELAGEM DE DADOS

7 TECNOLOGIAS UTILIZADAS E APIS

7.1 ANGULAR

Plataforma de uso livre para implementação de frontend utilizando HTML, CSS e JavaScript, baseada em TypeScript, linguagem que permite escrever JavaScript de maneira disciplinada e segura como em Java. Os aplicativos são arquitetados utilizando Componentes, cada um contendo arquivos HTML, CSS e Typescript (para os métodos).

7.2 NODE

Node (ou formalmente *Node.js*) é um ambiente em tempo de execução open-source (código aberto) e multiplataforma que permite aos desenvolvedores criarem todo tipo de aplicativos e ferramentas do lado servidor (backend) em JavaScript.

Tem implícito o framework Express para desenvolvimento Web e um Gerenciador de Pacotes (NPM, na sigla em inglês) que provê acesso a centenas de milhares de pacotes de uso livre, como o Mongoose para MongoDB.

7.3 MongoDB

Um Banco de Dados mais versátil que o relacional, atualmente muito utilizado para aplicativos Web. Em NodeJS pode ser utilizado com o pacote Mongoose, uma boa alternativa para um projeto bem organizado.

7.4 REST (Representational State Transfer)

É uma API (Application Programming Interface) independente de linguagem utilizada para comunicação padronizada com a camada de dados através do protocolo HTTP, que disponibiliza comandos semelhantes às operações em SQL. São eles:

- GET: obter uma representação de um recurso;
- POST: criar um novo recurso;
- PUT: criar novo recurso ou modificar um recurso já existente;
- DELETE: remover um recurso existente;

7.5 Visual Studio Code (VS Code)

Editor de código-fonte desenvolvido pela Microsoft para Windows, Linux e macOS. inclui suporte para depuração, controle Git, realce de sintaxe, complementação inteligente de código, snippets e refatoração de código.

7.6 Git/GitHub

Sistema de controle de versão em que podemos desenvolver projetos nos quais diversas pessoas contribuem simultaneamente editando, criando códigos e novos arquivos, permitindo que os mesmos possam existir sem o risco de suas alterações serem sobrescritas.

8 TELAS DO PROTÓTIPO EVOLUTIVO

9 RELATÓRIOS EMITIDOS

10 DOCUMENTOS GERADOS