Controle TL

Documento de Arquitetura de Software

1.0

Índice Analítico

[1. Introdução 3](#_Toc321036874)

[1.1 Finalidade 3](#_Toc321036875)

[1.2 Escopo 3](#_Toc321036876)

[1.3 Definições, Acrônimos e Abreviações 3](#_Toc321036877)

[1.4 Referências 3](#_Toc321036878)

[1.5 Visão Geral 3](#_Toc321036879)

[2. Representação Arquitetural 4](#_Toc321036880)

[3. Metas e Restrições da Arquitetura 4](#_Toc321036881)

[4. Visão de Casos de Uso 4](#_Toc321036882)

[4.1 Realizações de Casos de Uso 6](#_Toc321036883)

[5. Visão Lógica 7](#_Toc321036884)

[5.1 Visão Geral 7](#_Toc321036885)

[5.2 Pacotes des Design Significativos do Ponto de Vista da Arquitetura 8](#_Toc321036886)

[5.3 Diagrama de componentes](#_Toc321036887) 9

[6. Visão de Implantação 1](#_Toc321036889)0

[7. Visão de Dados 11](#_Toc321036891)

[8. Tamanho e Desempenho 11](#_Toc321036892)

[9. Qualidade 11](#_Toc321036893)

[10. Exceções 1](#_Toc321036894)2

Documento de Arquitetura de Software

# Introdução

Esse documento provê uma visão de alto nível dos objetivos da arquitetura, dos estilos arquiteturais e componentes que foram selecionados para implementação das funcionalidades propostas pelos casos de usos levantados do Controle TL.

## Finalidade

Este documento oferece uma visão geral arquitetural do sistema Controle TL, usando diversas visões arquiteturais para representar diferentes aspectos do sistema. O objetivo deste documento é capturar e comunicar as decisões arquiteturais significativas que foram tomadas em relação ao sistema.

## Escopo

Este Documento de Arquitetura de Software é aplicado ao Sistema Controle TL, que será desenvolvido por Guilherme Lima do curso de Sistemas de Informação da Universidade Federal de Sergipe como solução tecnológica durante o estágio.

## Definições, Acrônimos e Abreviações

Todas as definições, juntamente com os termos, acrônimos e abreviações necessárias à adequada interpretação e entendimento deste documento podem ser encontradas no documento de Glossário do Projeto.

## Referências

Os seguintes documentos foram utilizados como referência para a elaboração do documento arquitetura:

* Processos de Negócio
* Modelo de Casos de Uso
* *RationalUnifiedProcess*

## Visão Geral

A arquitetura utilizada é adaptada do próprio flutter (MVVM), com uma camada para as visualizações, uma camada para os modelos das visualizações, uma camada para os repositórios de acesso aos dados e uma camada para acesso aos dados.

# Representação Arquitetural

Este documento apresenta a arquitetura como uma série de visualizações, mencionadas acima.

Para representar a arquitetura do software foram utilizados como base os seguintes estilos arquiteturais:

* Camadas;
* Repositório de banco de dados.

# Metas e Restrições da Arquitetura

Para que o software tenha um comportamento esperado pelos stakeholders ele deve seguir as seguintes restrições:

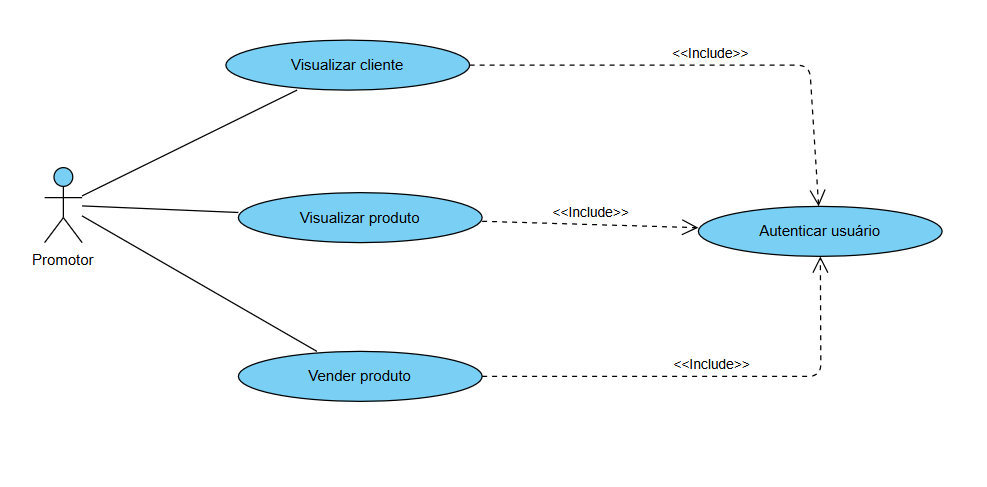
* A versão Mobile do Controle TL dará suporte para execução apenas em sistemas Android.

# Visão de Casos de Uso

Nessa seção serão listados os casos de uso que são representados no modelo de casos de uso. Esses casos de uso são:

* CSU01 - Autenticar Usuário;
* CSU02 – Visualizar Cliente;
* CSU03 – Visualizar Produto;
* CSU04 – Vender Produto;

Figura 1: Diagrama de Casos de Uso



Fonte: Autoria própria.

## Realizações de Casos de Uso

A descrição de cada caso de uso contido no diagrama da Figura 1 encontra-se detalhada no diretório Requisitos > Casos de Uso.

# Visão Lógica

## Visão Geral

A visão lógica do Controle TL é composta principalmente por 6 pacotes:

* UI: Esse pacote representa a implementação da apresentação, a lógica da apresentação, widgets compartilhados, loggers e temas de cores do sistema.
* Config: Neste pacote, todas as configurações e escuta de canais de transmissão do supabase é realizada, além disso, a configuração do gerenciamento de estado do sistema também é feita através dos providers.
* Data: Nesse pacote de acesso aos dados é onde está os serviços para o acesso ao supabase, além de sua camada intermediária, as implementações dos repositórios.
* Domain: Neste pacote está os contratos dos repositórios e os modelos das entidades que estão sendo representadas no banco de dados.
* Routing: Este pacote disponibiliza todas as rotas do sistema.
* Utils: Neste pacote está os utilitários, constantes globais e validações de entrada de dados.

# Visão de Dados (Parcial)

A base de dados utilizada encontra-se no Supabase (BaaS) e a utilização do banco postreSql desse back-end é feita através dos seguintes objetos:

* polibras\_pedcab: Nesta tabela é feito o inserte do cabeçalho do pedido.
* polibras\_pedcorp: Nesta tabela é feito o inserte do corpo do pedido.
* vw\_merchandising\_clientes: Esta view é utilizada para obter todos os clientes do usuário logado, onde, possui a regra de negócio.
* vw\_merchandising\_preco: Esta view é utilizada para obter os preços de venda dos produtos, onde, possui a regra de negócio.
* vw\_merchandising\_produtos: Esta view é utilizada para obter os produtos, onde, possui a regra de negócio.

Em resumo, o fluxo de operações segue do seguinte modo:

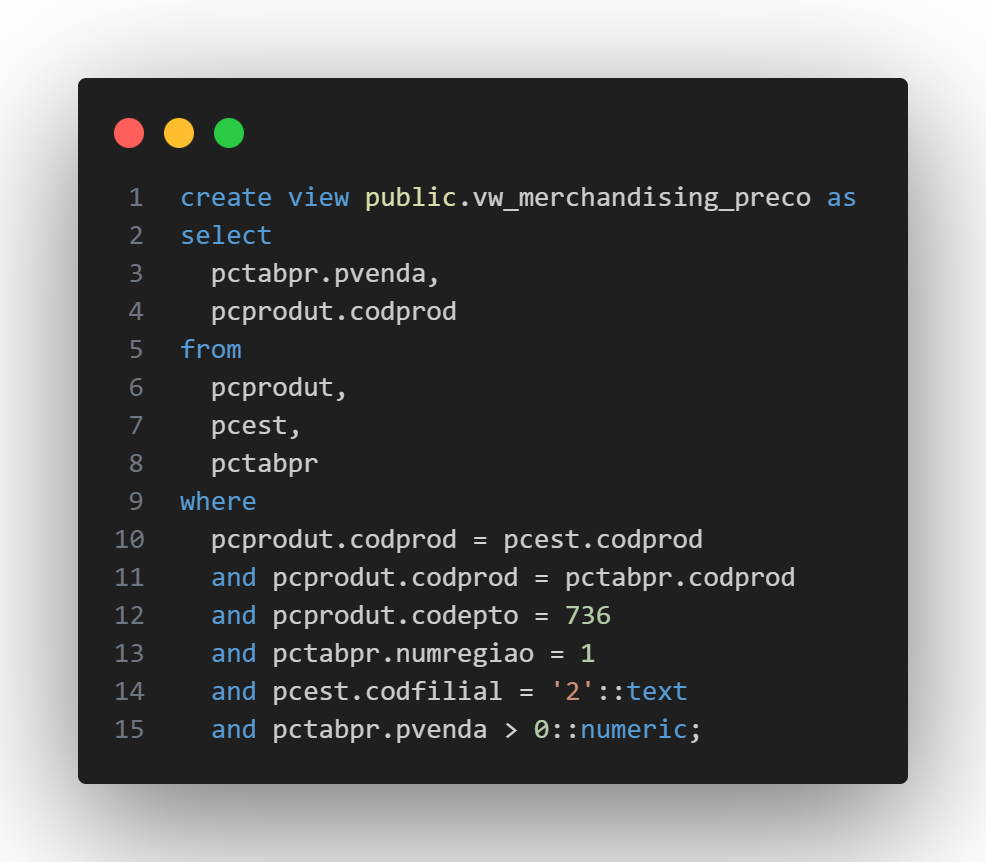
Inserção do cabeçalho do pedido > Inserção do corpo do pedido > Atualização da coluna importado para ‘9’ onde acaba indicando ao PostgreSQL para levar as mudanças para o Oracle.

Figura 2: View de clientes.



Fonte: Autoria própria.

Figura 3: View de preço de venda.



Fonte: Autoria própria.

Figura 4: View de produtos.



Fonte: Autoria própria.

# Tamanho e Desempenho

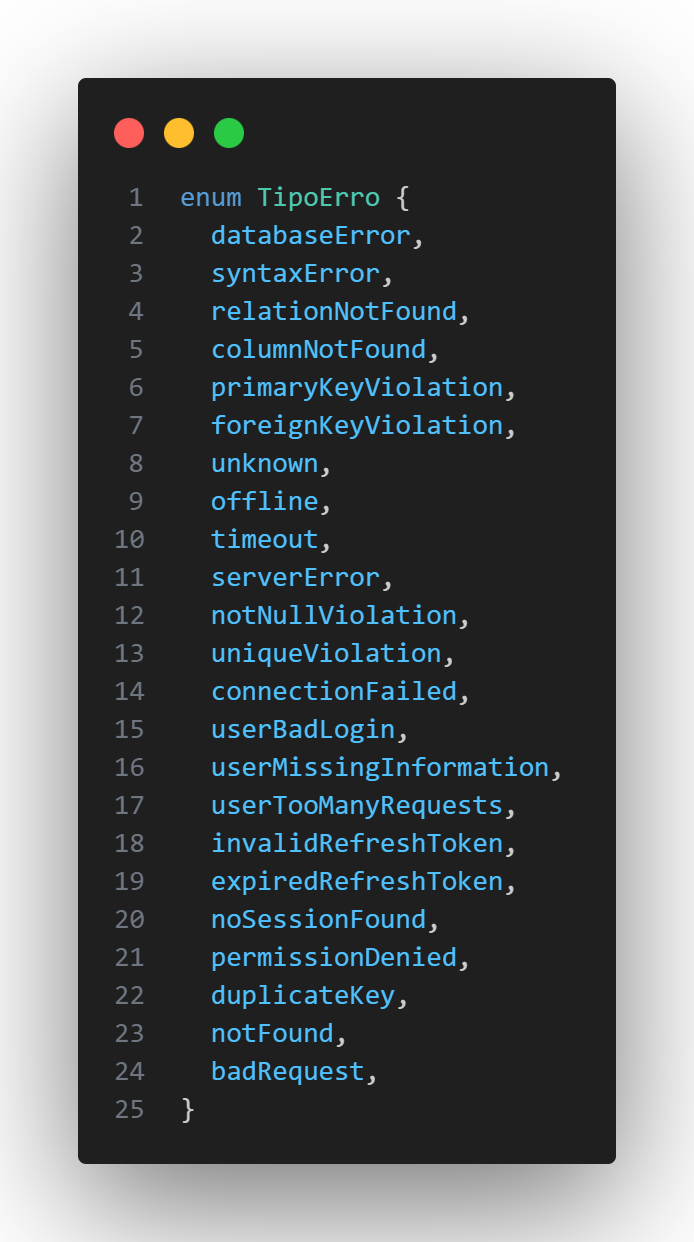
Aplicativo escalável com comunicação ao back-end do Supabase, onde, por sua vez comunica-se com o banco Oracle da empresa. A expansividade do sistema móvel depende somente dessa comunicação.

# Qualidade

O projeto adota boas práticas de desenvolvimento e está aberto a melhorias e refatorações.

# Exceções

Figura 5: Tipos de exceções conhecidas.



Fonte: Autoria própria.