



CENTRO UNIVERSITÁRIO DINÂMICA DAS CATARATAS
Sistemas de Informação - Prática Profissional

Sistema Móveis Três Lagoas Especificação de Arquitetura

Versão 1.0.2

Histórico da Revisão

Data	Versão	Descrição	Autor
03 de agosto de 2016	1.0.0	Versão inicial	Ricardo Augusto
05 de agosto de 2016	1.0.1	Versão incrementada	Ricardo Augusto
30 de novembro de 2016	1.0.2	Versão final	Ricardo Augusto

Índice Analítico

1. Introdução.....	4
1.1 Finalidade.....	4
1.2 Definições, Acrônimos e Abreviações.....	4
2. Visão geral da arquitetura.....	4
2.1 Visão de Caso de Uso	4
2.1.1 Manter Clientes.....	5
2.1.2 Manter fornecedores	6
2.1.3 Manter produtos.....	6
2.1.4 Manter funcionários.....	7
2.1.5 Manter usuários.....	7
2.1.6 Manter vendas.....	8
2.1.7 Manter compras	9
2.1.8 Manter contas a pagar	9
2.1.9 Manter contas a receber	9
2.1.10 Manter caixa.....	10
2.2 Visão Lógica.....	11
2.3 Visão de Processos.....	11
3. Componentes físicos do projeto	12
3.1 Model.....	12
3.2 View	12
3.3 Controller.....	12
3.4 DAO	12
3.5 SGDB	12
3.6 Relatório	12

1. Introdução

1.1 Finalidade

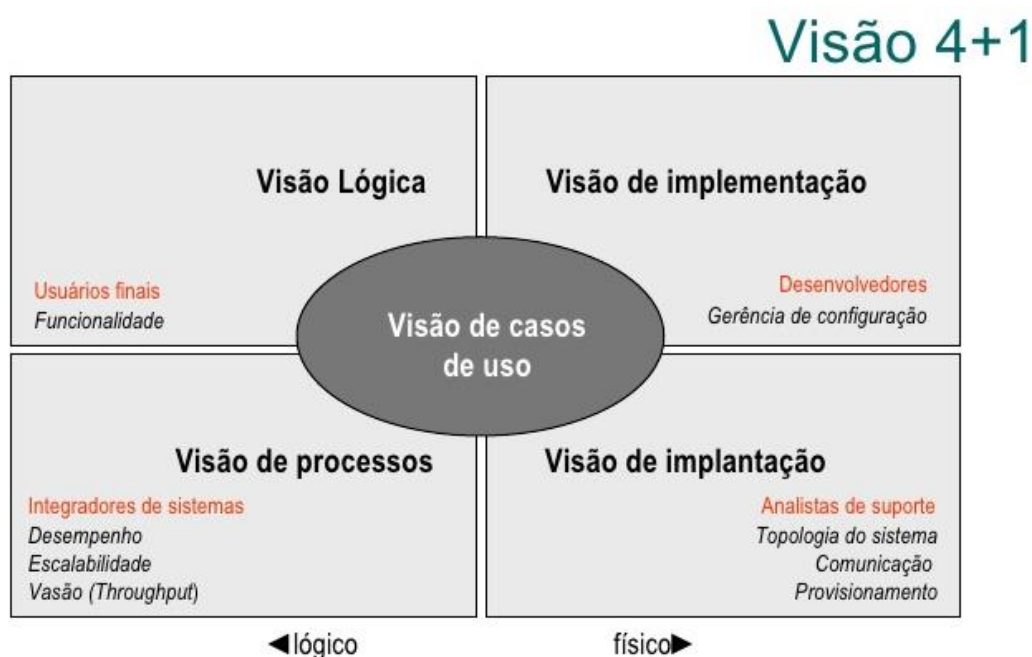
Este documento apresenta uma visão geral abrangente da arquitetura do sistema e utiliza uma série de visões arquiteturais diferentes para ilustrar os diversos aspectos do sistema. Sua intenção é capturar e transmitir as decisões significativas do ponto de vista da arquitetura que foram tomadas em relação ao sistema.

1.2 Definições, Acrônimos e Abreviações

SMTL – Sistema Móveis Três Lagoas

2. Visões da arquitetura

Este documento apresenta a arquitetura do sistema através das seguintes visões arquiteturais:

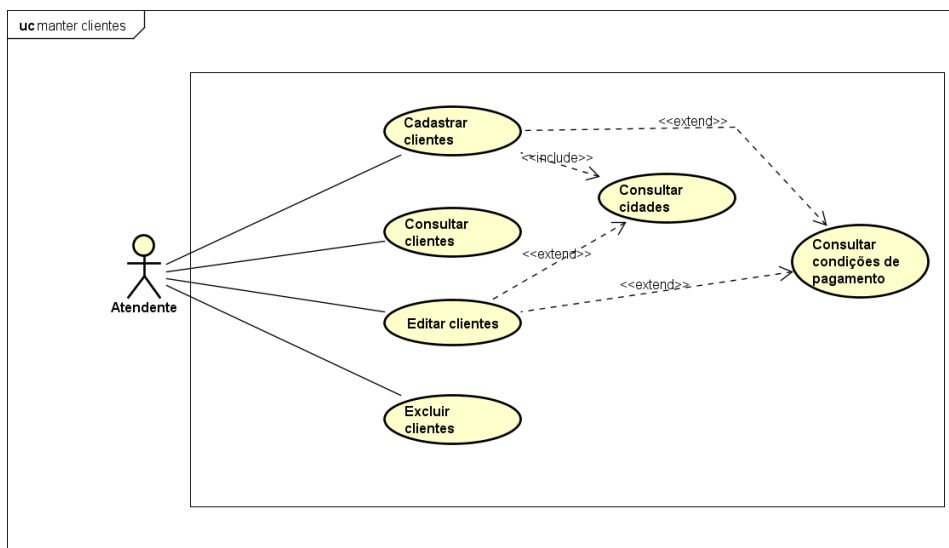


2.1 Visão de Caso de Uso

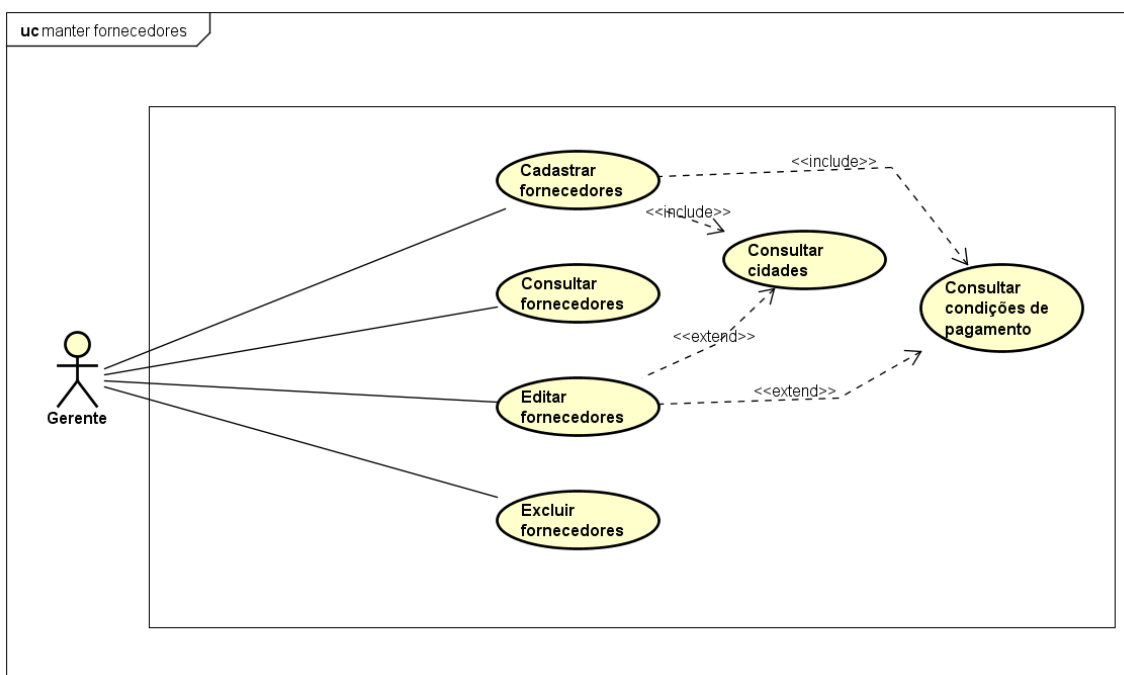
O intenção desta visão é apresentar os principais casos de uso para a arquitetura do sistema como um todo.

Caso de Uso	Impacto
Manter Clientes	Caso de Uso responsável pela inclusão, alteração e exclusão de clientes.
Manter Fornecedores	Caso de Uso responsável pela inclusão, alteração e exclusão de fornecedores.
Manter Produtos	Caso de Uso responsável pela inclusão, alteração e exclusão de produtos.
Manter Funcionários	Caso de Uso responsável pela inclusão, alteração e exclusão de funcionários.
Manter Usuários	Caso de Uso responsável pela inclusão, alteração e exclusão de usuários.
Manter Vendas	Caso de Uso responsável pela inclusão, alteração e exclusão de vendas.
Manter Compras	Caso de Uso responsável pela inclusão, alteração e exclusão de compras.
Manter Contas a pagar	Caso de Uso responsável pela inclusão, alteração, exclusão e baixa das contas a pagar.
Manter contas a receber	Caso de Uso responsável pela inclusão, alteração, exclusão e baixa das contas a receber.
Manter caixa	Caso de Uso responsável pela manipulação do caixa local.

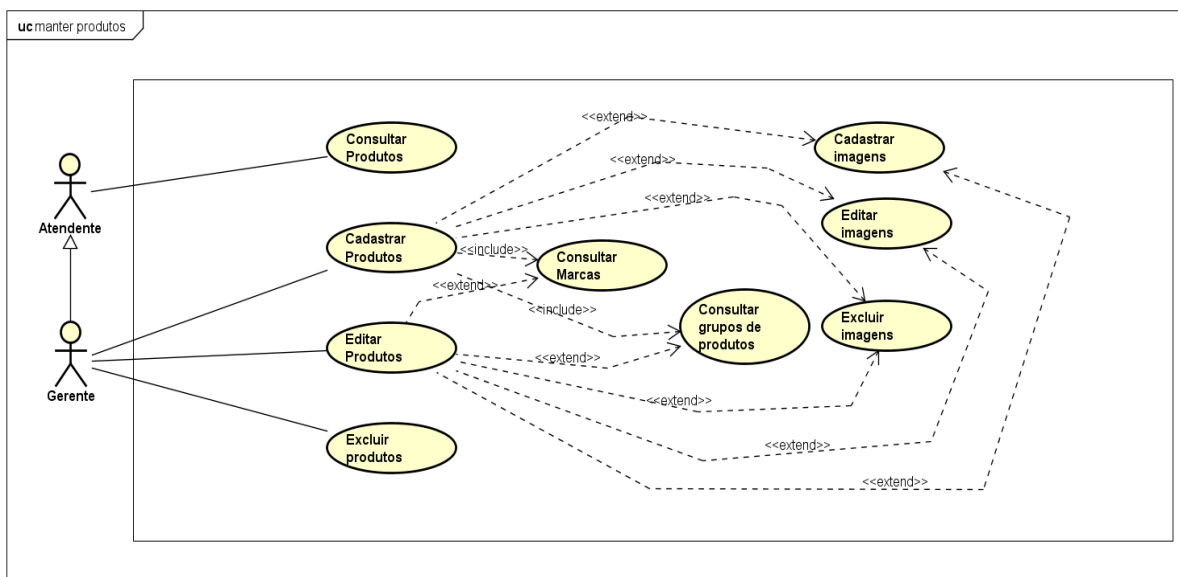
2.1.1 Manter Clientes



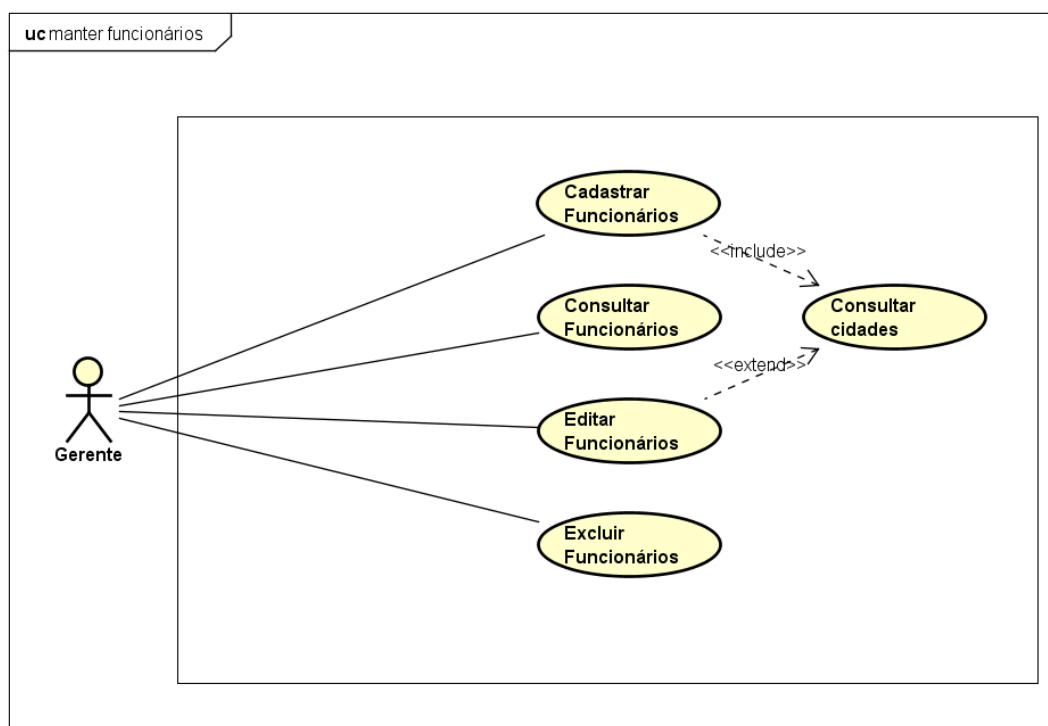
2.1.2 Manter Fornecedores



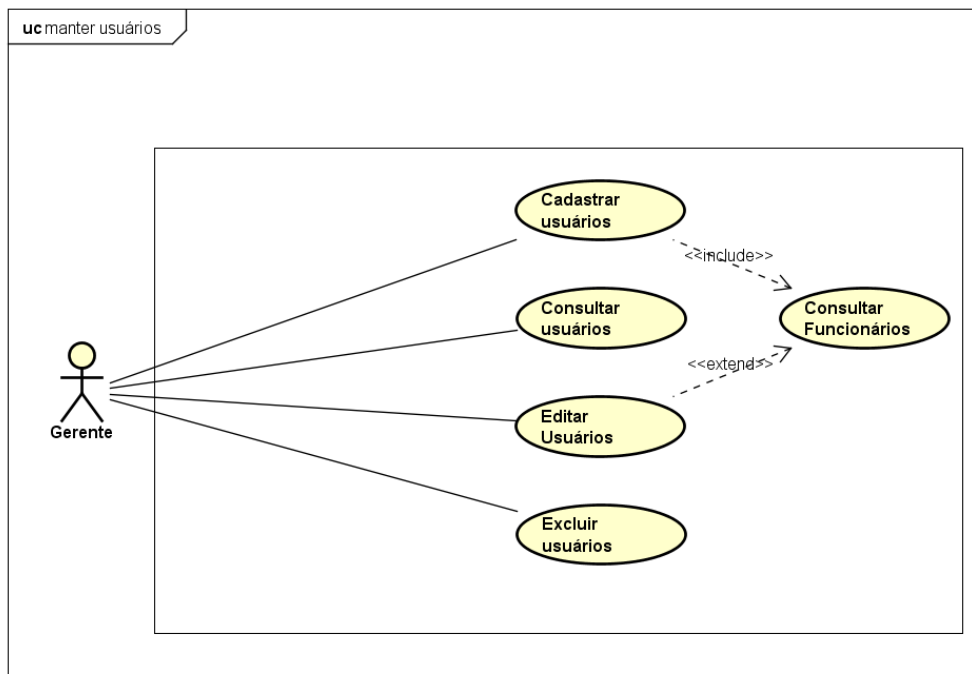
2.1.3 Manter Produtos



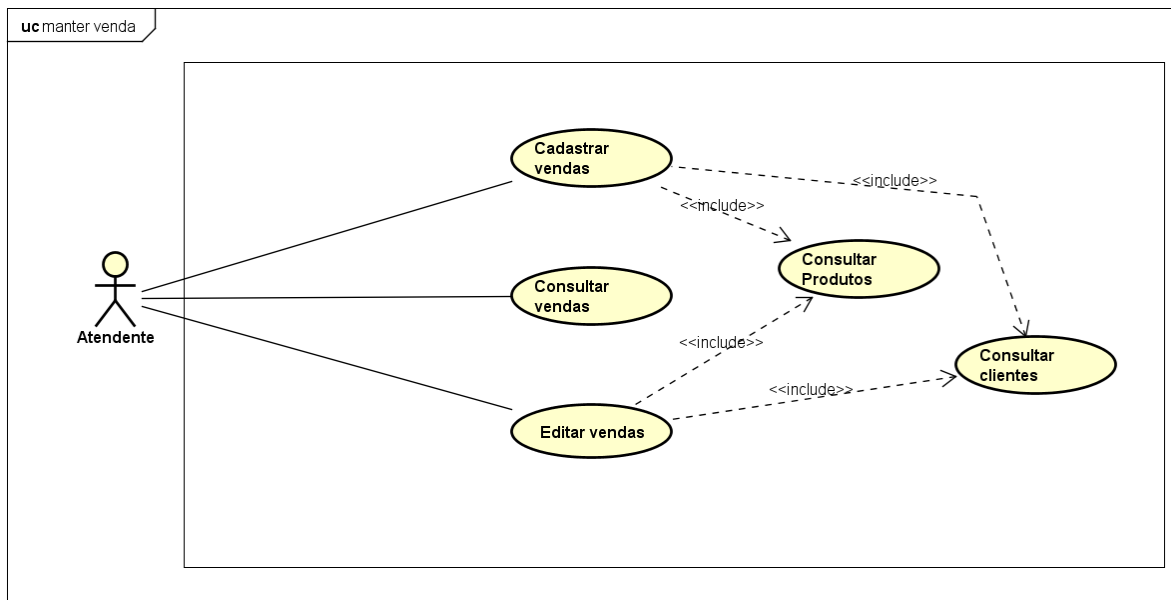
2.1.4 Manter Funcionários



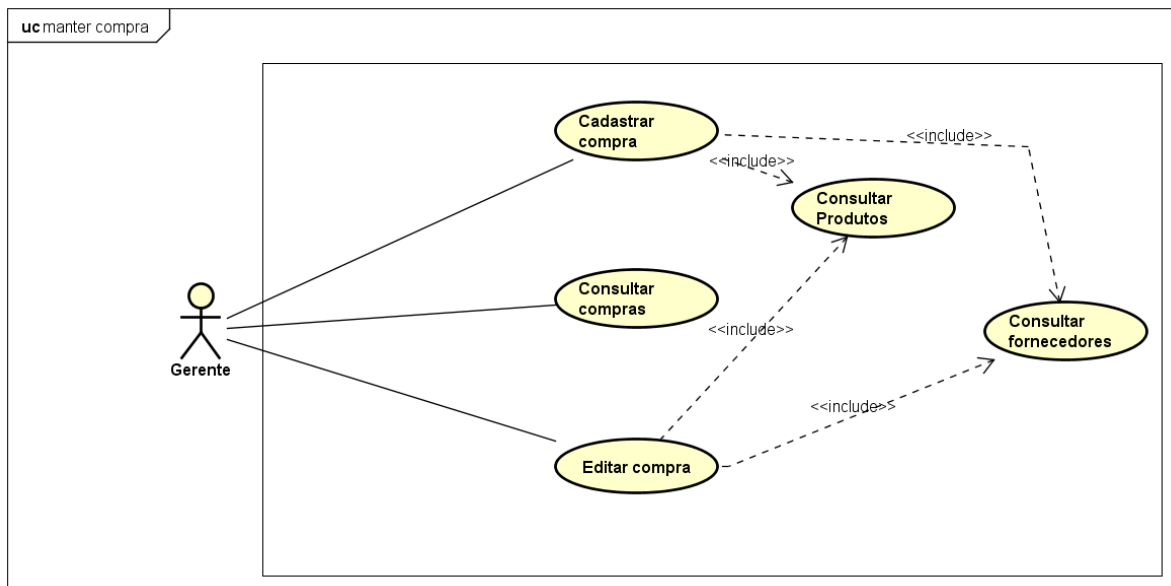
2.1.5 Manter Usuários



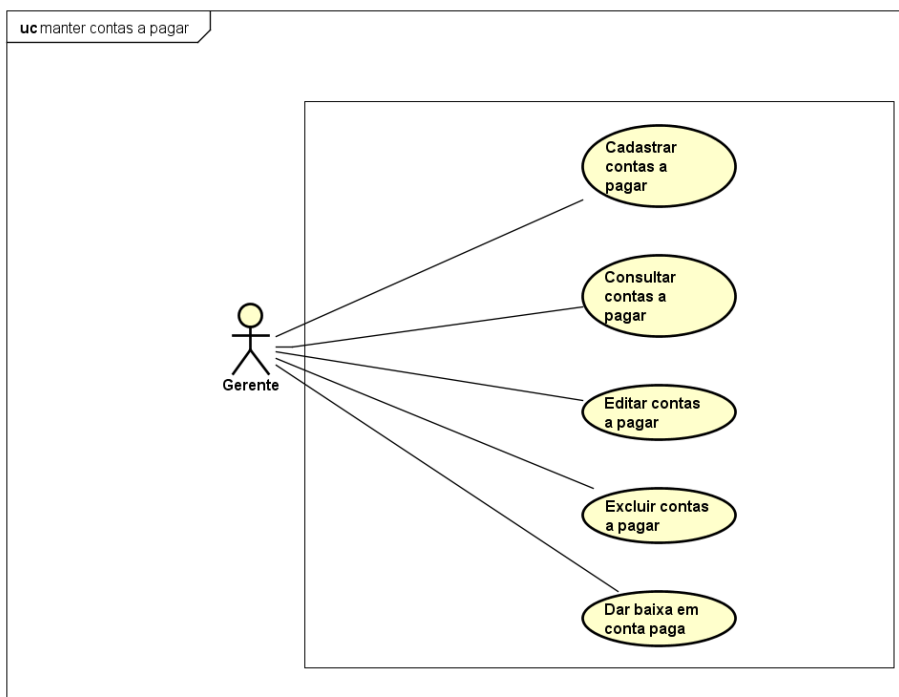
2.1.6 Manter Vendas



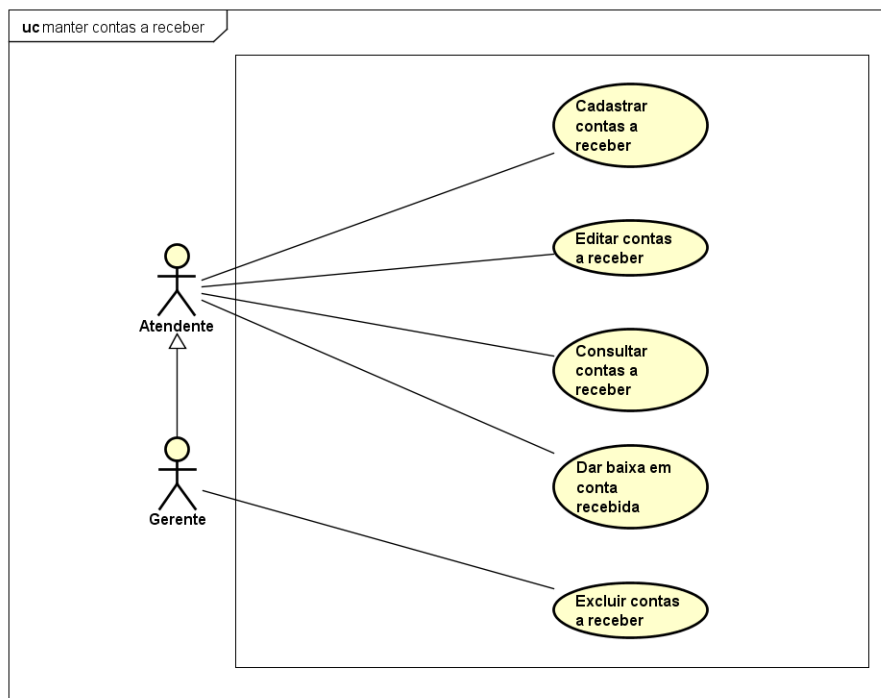
2.1.7 Manter Compras



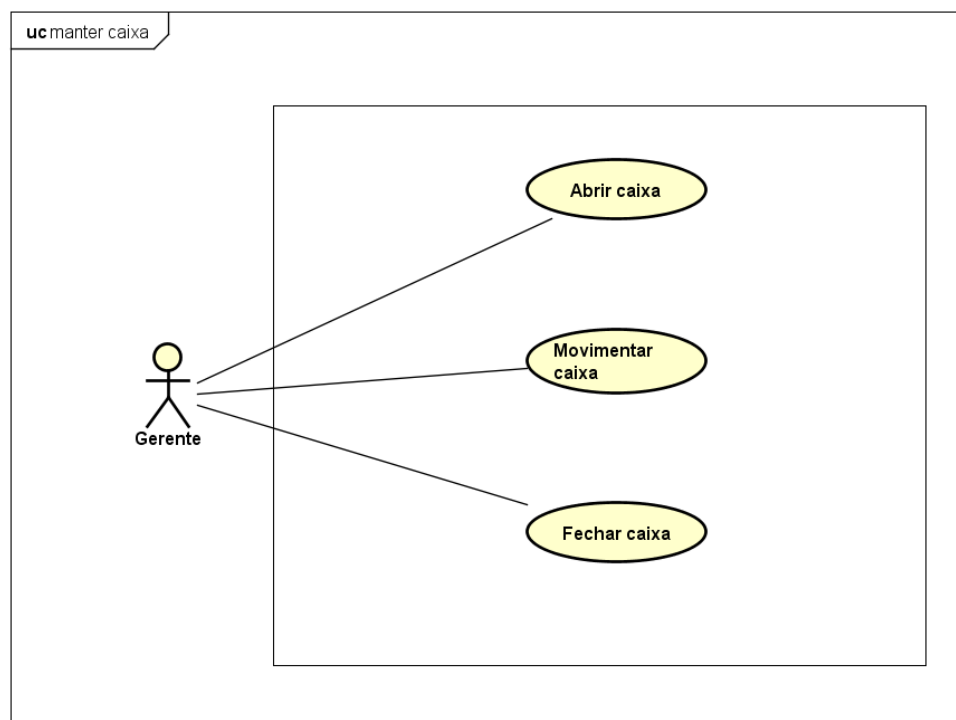
2.1.8 Manter Contas a pagar



2.1.9 Manter Contas a receber



2.1.10 Manter Caixa



2.2 Visão Lógica

O SMTL será desenvolvido em Delphi 7 com Banco de Dados MySQL, utilizando o componente MyDac para integração Aplicação-Banco de Dados, e o DBForge para gerenciamento da base de dados. Para geração de relatórios será utilizado o QuickReport.

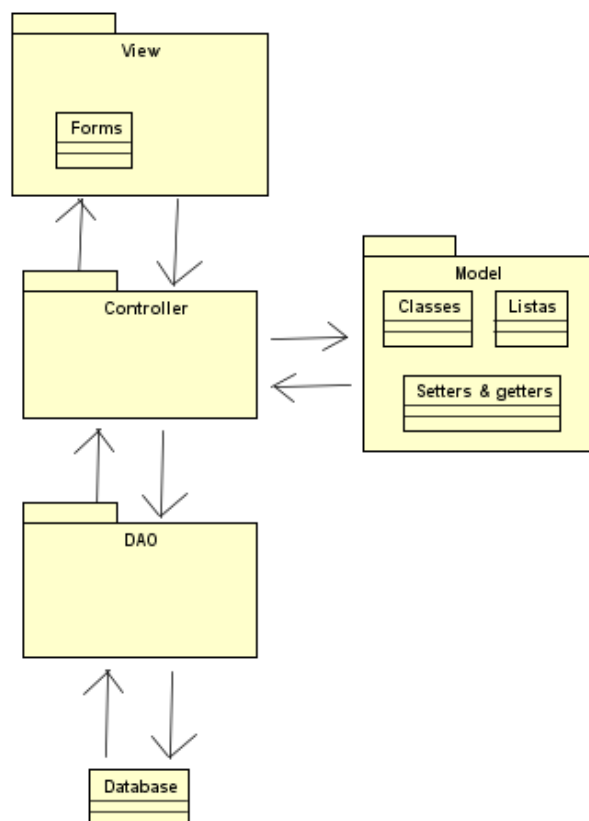
A arquitetura será definida no estilo “MVC” (Model-View-Controller), onde:

- “Model” é a camada modelo, onde estão as classes de objetos;
- “View” é a camada de apresentação e;
- “Controller” é a camada de negócio, responsável por implementar as regras de negócio referentes às consultas, inclusões, exclusões e edições no sistema.

Além destas camadas, há a camada de persistência, de comunicação com a base de dados, a DAO (Data Access Object).

2.3 Visão de Processos

O diagrama abaixo representa a visão geral da arquitetura do sistema adotada para o projeto:



3. Componentes Físicos do Projeto

3.1 Model

Componente	Descrição	Versão	Benefícios
Class – Units Delphi	Classes do sistema: Conceito de orientação a objetos	7	Fácil desenvolvimento e manutenção

3.2 View

Componente	Descrição	Versão	Benefícios
Forms	Forms nativos do Delphi	7	Fácil manutenção e usabilidade boa

3.3 Controller

Componente	Descrição	Versão	Benefícios
Units Delphi	Componentes nativos do Delphi	7	Fácil desenvolvimento e manutenção

3.4 DAO

Componente	Descrição	Versão	Benefícios
Units Delphi	Componentes nativos do Delphi	7	Fácil desenvolvimento e manutenção
MyDAC	Componente de comunicação com DB	7	Fácil desenvolvimento e manutenção

3.5 SGDB

Componente	Descrição	Versão	Benefícios
DBForge	Software de gerenciamento de DB	Studio Express	Fácil manutenção
MySQL	DataBase	6.5	Fácil manutenção

3.6 Relatório

Componente	Descrição	Versão	Benefícios
QuickReport	Componente que gera relatórios		Fácil manutenção