

1º Desafio de Análise e Desenvolvimento de Software
Instituto Federal de Goiás – Campus Uruaçu

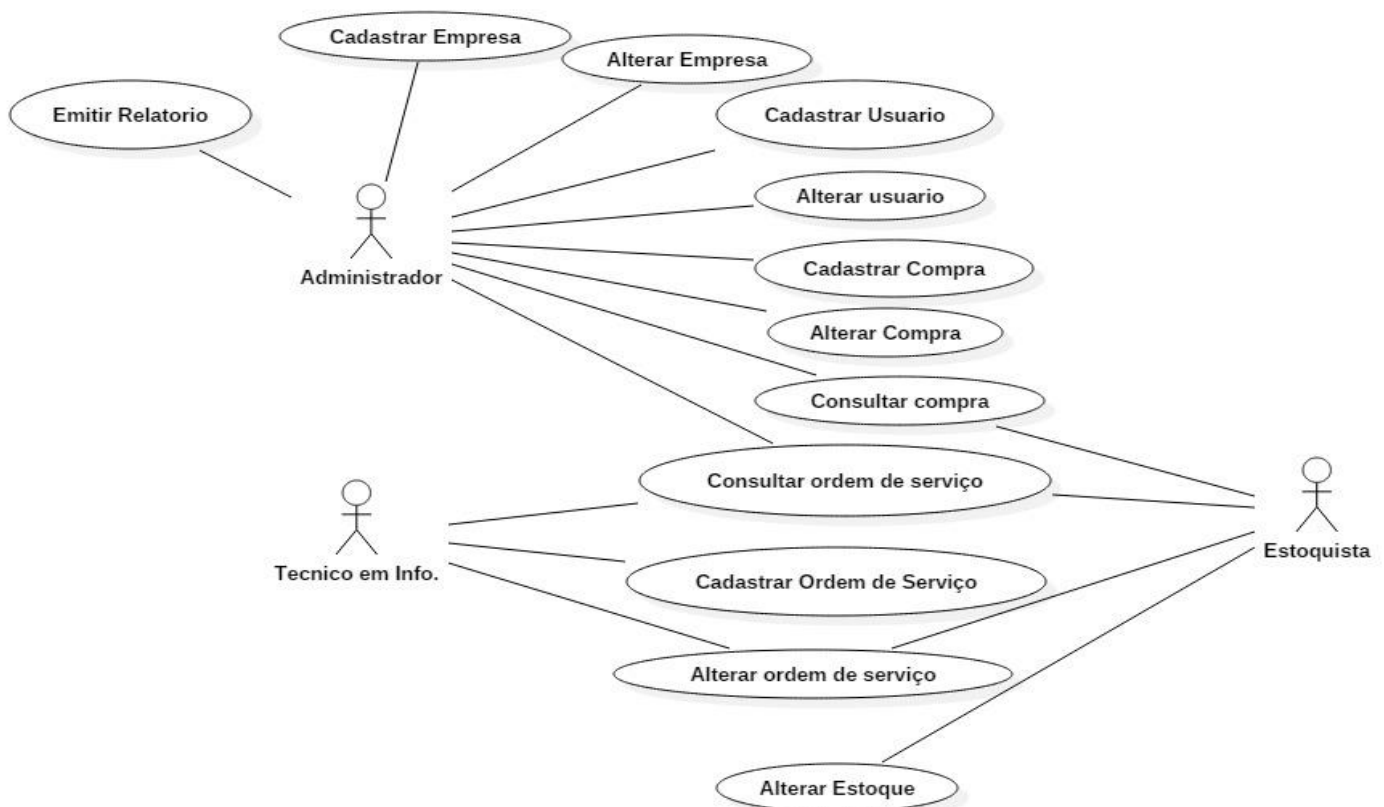
ProDev

- **Caio Silva**
- **Dennys Fonseca**
- **Kevyn Carlos**

Requisitos.

- Gerenciamento de entrada e saída de produtos e ferramentas do estoque.
- Acesso a informações precisas sobre quantidade de produtos no estoque.
- Controle de permissões dos usuários de partir do seu tipo. (administrador, estoquista, técnico em info. e etc...)
- Gerenciamento de ordens de serviço
- Controle de gastos feitos em compras. (semana, mês, ano)
- Controle de ferramentas/produtos que não estão sendo usados, porém estão no estoque.
- Emissão de relatório por tipo ordem de serviço feita pelo o colaborador.
- Emissão de relatório de quantidade de compras feitas por fornecedor.
- Emissão de relatório de estoque mínimo e máximo a partir do tipo do produto.
- Sistema multiempresa. (Implementação em filiais ou afins).

Diagrama de caso de uso.



Especificação de caso de uso.

Identificação: 001	
Nome: Emitir Relatório	
Atores: Administrador	
Sequencia típica de eventos	
Ator	Sistema
1. Escolhe tipo de relatório.	4. Buscas informações no banco.
2. Faz a filtragem.	5. Emite o relatório.
3. Clica no botão Pesquisar.	
Sequencia alternativa	
2a. Opções p/ filtrar não condiz com o relatório.	4a. Informações não condizem com o tipo do relatório.
1. Verificar se o relatório escolhido é o que necessita.	1. Verificar como informações foram cadastradas.
	5a. Dados incoerentes.
	1. Entrar em contato com adm do sistema

Identificação: 002	
Nome: Cadastrar Empresa.	
Atores: Administrador	
Sequencia típica de eventos	
Ator	Sistema
1. Preencher formulário.	3. Registra informações no banco.
2. Clica no botão Cadastrar.	
Sequencia alternativa	
1a. Tipo de informação invalida.	3a. Erro ao realizar operação.
1. Avaliar o tipo de informação que está sendo inserida	1. Entrar em contato com adm do sistema.

Identificação: 003	
Nome: Alterar Empresa.	
Atores: Administrador	
Sequencia típica de eventos	
Ator	Sistema
1. Pesquisar por Empresa	4. Traz formulário com as informações da empresa escolhida.
2. Escolher o Empresa.	7. Salva informações no banco de dados.
3. Clicar no botão editar.	
5. Edita informações da empresa.	
6. Clica no botão salvar.	
Sequencia alternativa	
1a. traz informações que não são da empresa.	4a. Formulário faltando informações.
1. Entrar em contato com adm do sistema. 2. Verificar se a empresa escolhida é acerta.	1. Entrar em contato com adm do sistema. 2. Verificar se a empresa escolhida é acerta.
7a. Erro ao realizar operação.	
1. Entrar em contato com adm do sistema.	

Identificação: 004	
Nome: Cadastrar Usuario.	
Atores: Administrador	
Sequencia típica de eventos	
Ator	Sistema
1. Preencher formulário.	3. Registra informações no banco.
2. Clica no botão Cadastrar.	
Sequencia alternativa	
1a. Tipo de informação invalida.	3a. Erro ao realizar operação.
1. Avaliar o tipo de informação que está sendo inserida	1. Entrar em contato com adm do sistema.

Identificação: 005	
Nome: Alterar usuário.	
Atores: Administrador	
Sequencia típica de eventos	
Ator	Sistema
1. Pesquisar por usuário	4. Traz formulário com as informações do usuário escolhido.
2. Escolher o usuário.	7. Salva informações no banco de dados.
3. Clicar no botão editar.	
5. Edita informações do usuário.	
6. Clica no botão salvar.	
Sequencia alternativa	
1a. traz informações que não são da empresa.	4a. Formulário faltando informações.
1. Entrar em contato com adm do sistema. 2. Verificar se a empresa escolhida é certa.	3. Entrar em contato com adm do sistema. 4. Verificar se a empresa escolhida é certa.
7a. Erro ao realizar operação.	
2. Entrar em contato com adm do sistema.	

Identificação: 006	
Nome: Cadastrar compra.	
Atores: Administrador	
Sequencia típica de eventos	
Ator	Sistema
1. Preencher formulário.	3. Registra informações no banco.
2. Clica no botão Cadastrar.	
Sequencia alternativa	
1a. Tipo de informação invalida.	3a. Erro ao realizar operação.
1. Avaliar o tipo de informação que está sendo inserida	1. Entrar em contato com adm do sistema.

Identificação: 007	
Nome: Alterar compra.	
Atores: Administrador	
Sequencia típica de eventos	
Ator	Sistema
1. Pesquisar por compra	4. Traz formulário com as informações da compra escolhida.
2. Escolher o compra.	7. Salva informações no banco de dados.
3. Clicar no botão editar.	
5. Edita informações da empresa.	
6. Clica no botão salvar.	
Sequencia alternativa	
1a. traz informações que não são da empresa.	4a. Formulário faltando informações.
1. Entrar em contato com adm do sistema. 2. Verificar se a empresa escolhida é certa.	1. Entrar em contato com adm do sistema. 2. Verificar se a empresa escolhida é certa.
7a. Erro ao realizar operação.	
1. Entrar em contato com adm do sistema.	

Identificação: 008	
Nome: Consultar compra.	
Atores: Administrador, Estoquista	
Sequencia típica de eventos	
Ator	Sistema
1. Pesquisar por compra	4.Traz tabela com as informações da compra escolhida.
2. Escolher o compra.	7. Salva informações no banco de dados.
3. Clicar no botão de visualizar.	
5. Edita informações da empresa.	
6. Clica no botão salvar.	
Sequencia alternativa	
1a. traz informações que não são da empresa.	
2. Entrar em contato com adm do sistema.	
3. Verificar se a empresa escolhida é acerta.	

Identificação: 009	
Nome: Consultar ordem de serviço.	
Atores: Administrador, Estoquista, Téc. Info.	
Sequencia típica de eventos	
Ator	Sistema
1. Pesquisar por ordem de serviço	4.Traz tabela com as informações da ordem de serviço escolhida.
2. Escolhe a ordem de serviço.	
3. Clicar no botão de visualizar.	
5. Visualiza as informações.	
Sequencia alternativa	
1a. traz informações que não são da ordem de serviço.	
1. Entrar em contato com adm do sistema.	
2. Verificar se a ordem de serviço escolhida é acerta.	

Diagrama de Banco de Dados

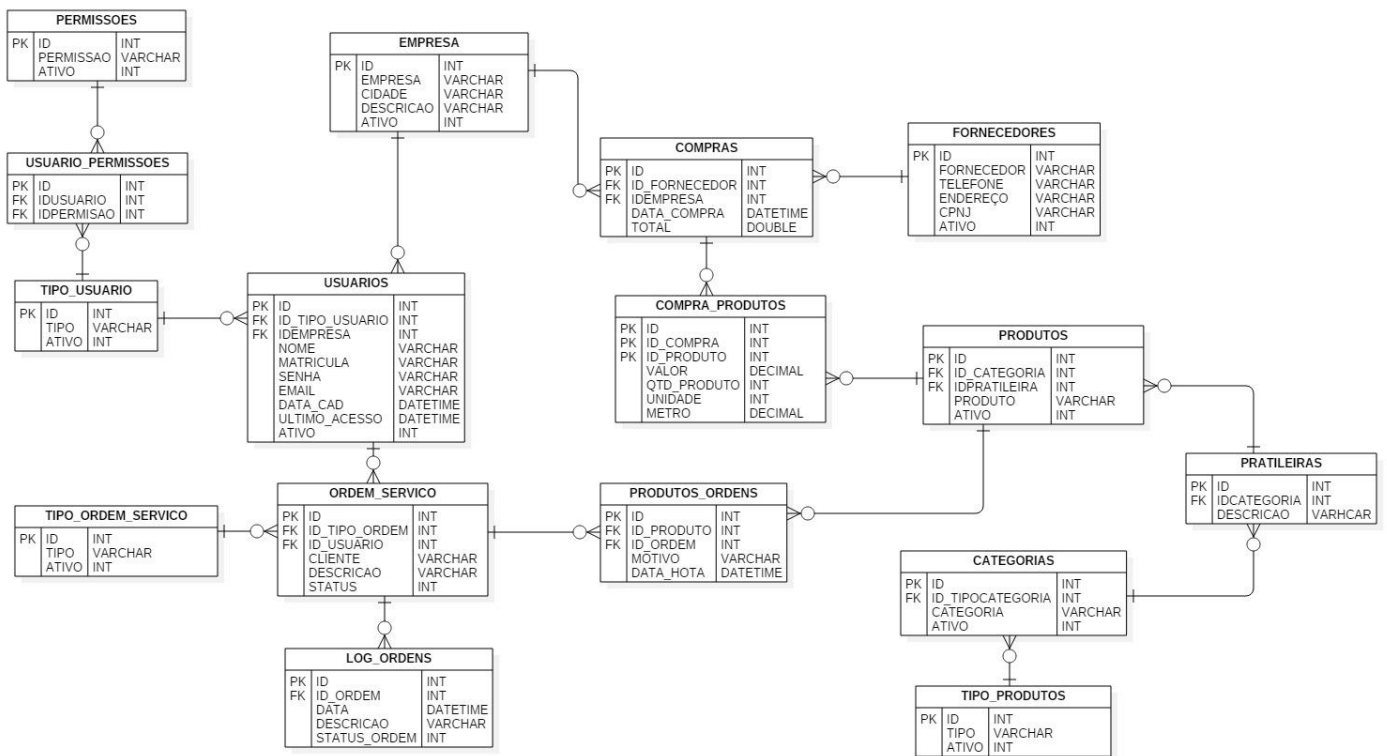
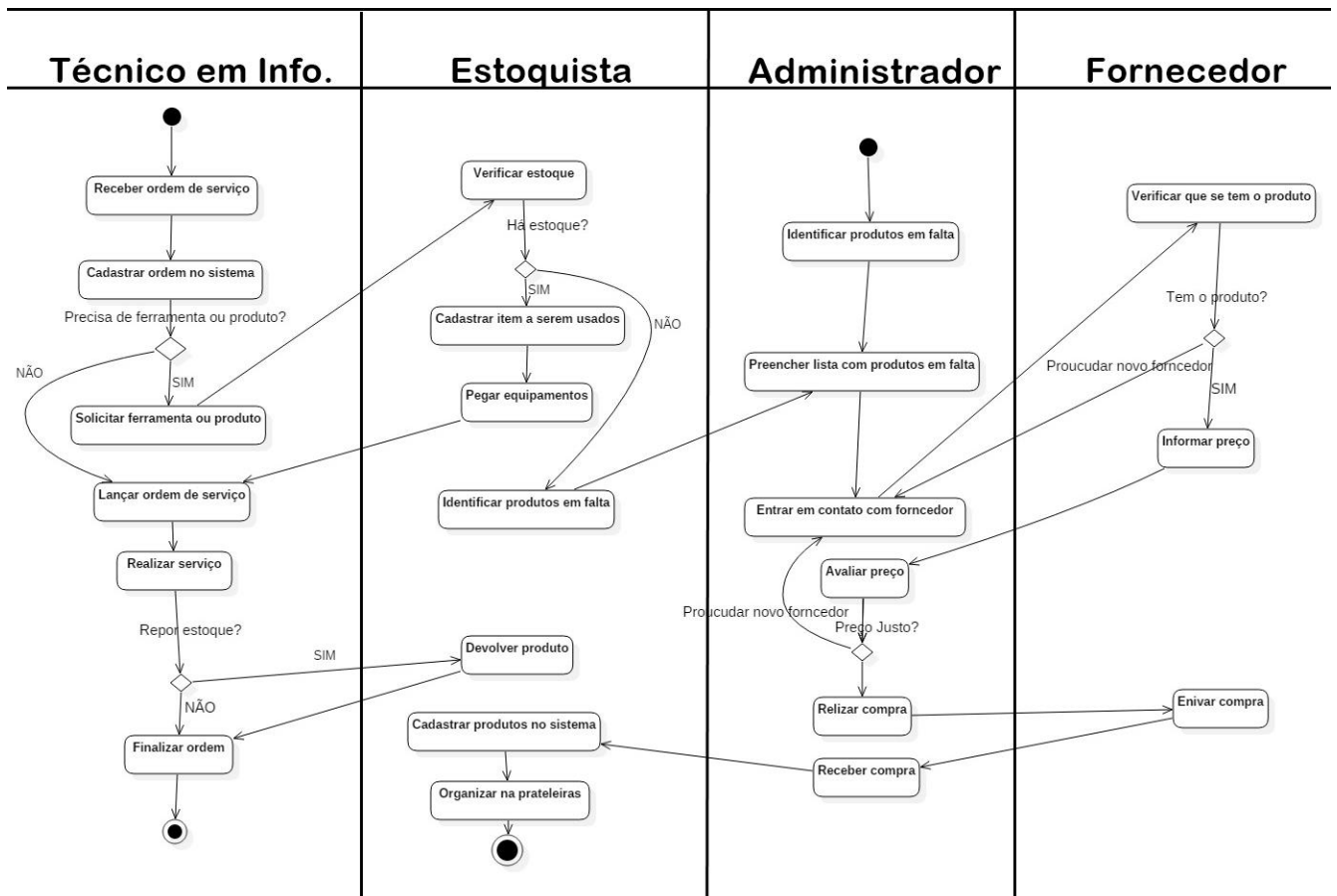
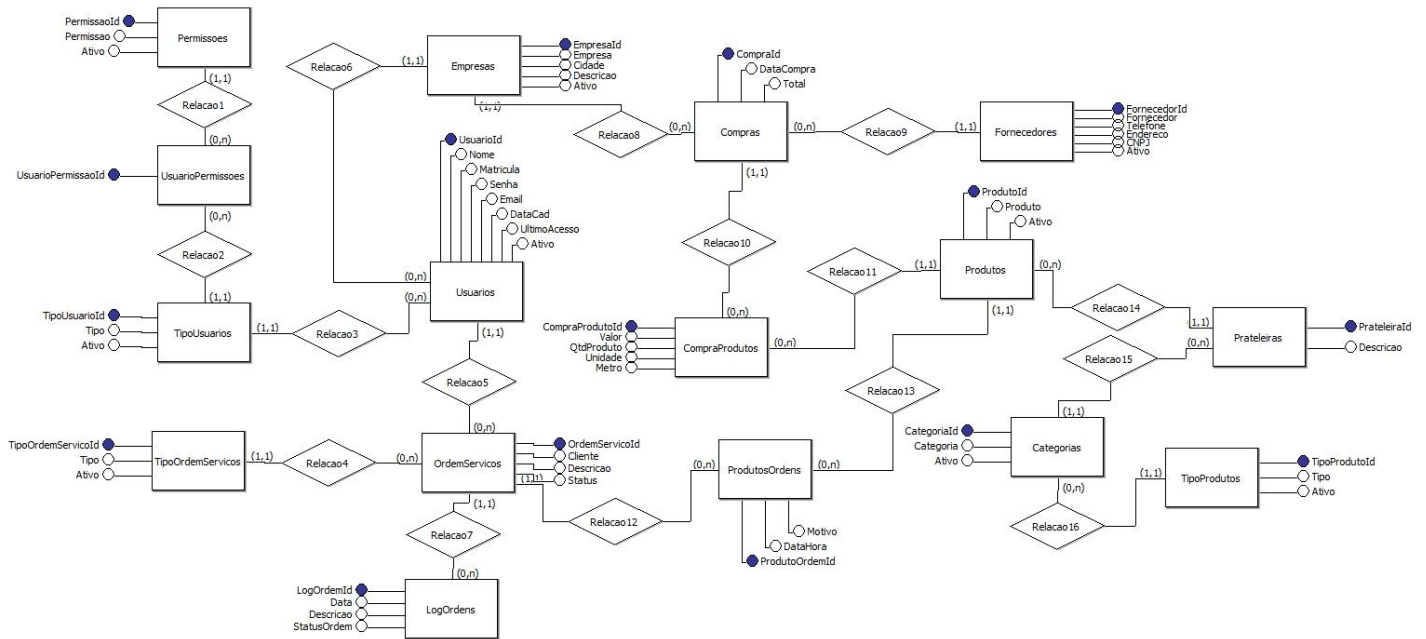


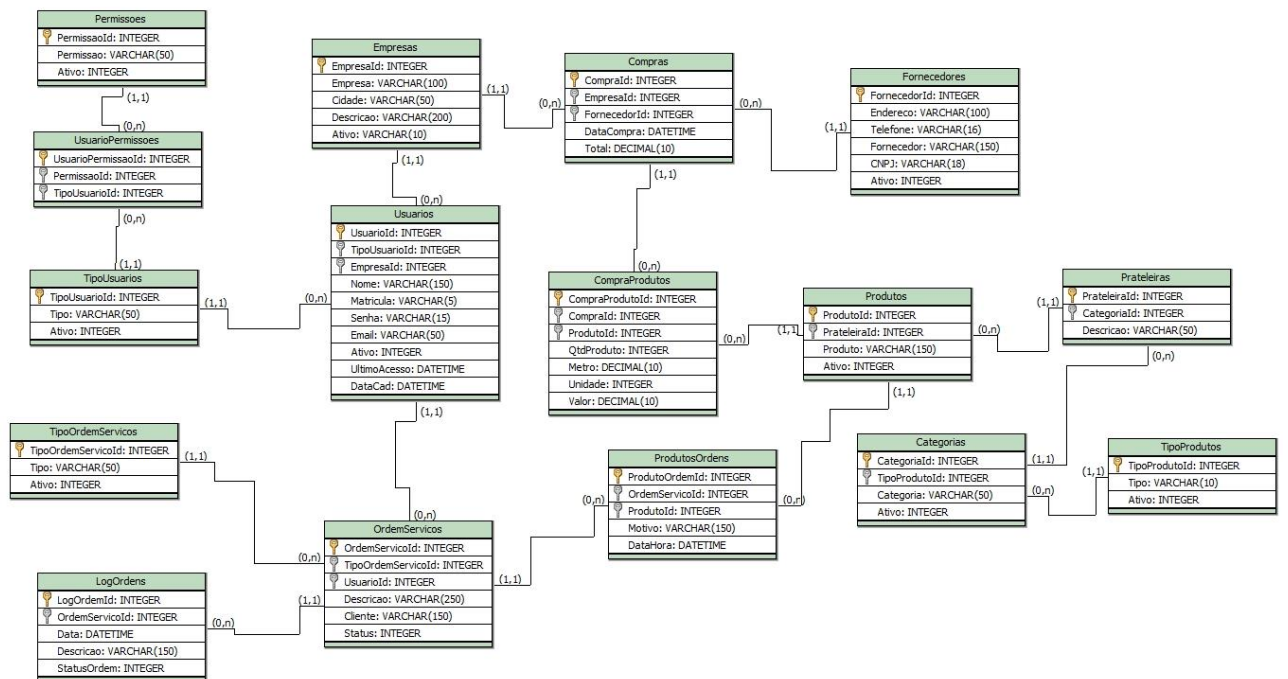
Diagrama de Atividades



Modelo Conceitual



Modelo Lógico



Modelo Físico

```
CREATE TABLE Permissoes (  
  PermissaoId INTEGER PRIMARY KEY,  
  Permissao VARCHAR(50),  
  Ativo INTEGER  
)
```

```
CREATE TABLE Empresas (  
  EmpresaId INTEGER PRIMARY KEY,  
  Empresa VARCHAR(100),  
  Cidade VARCHAR(50),  
  Descricao VARCHAR(200),  
  Ativo VARCHAR(10)  
)
```

```
CREATE TABLE ProdutosOrdens (  
  ProdutoOrdemId INTEGER PRIMARY KEY,  
  OrdemServicoId INTEGER,  
  ProdutoId INTEGER,  
  Motivo VARCHAR(150),  
  DataHora DATETIME  
)
```

```
CREATE TABLE UsuarioPermissoes (  
  UsuarioPermissaoId INTEGER PRIMARY KEY,  
  PermissaoId INTEGER,  
  TipoUsuarioId INTEGER,  
  FOREIGN KEY(PermissaoId) REFERENCES Permissoes (PermissaoId)  
)
```

```
CREATE TABLE TipoUsuarios (  
  TipoUsuarioId INTEGER PRIMARY KEY,  
  Tipo VARCHAR(50),  
  Ativo INTEGER  
)
```

```
CREATE TABLE TipoOrdemServicos (  
  TipoOrdemServicoId INTEGER PRIMARY KEY,  
  Tipo VARCHAR(50),  
  Ativo INTEGER  
)
```

```
CREATE TABLE Usuarios (  
  UsuarioId INTEGER PRIMARY KEY,  
  TipoUsuarioId INTEGER,  
  EmpresaId INTEGER,  
  Nome VARCHAR(150),  
  Matricula VARCHAR(5),  
  Senha VARCHAR(15),  
  Email VARCHAR(50),  
  Ativo INTEGER,  
  UltimoAcesso DATETIME,  
  DataCad DATETIME,  
  FOREIGN KEY(TipoUsuarioId) REFERENCES TipoUsuarios (TipoUsuarioId),  
  FOREIGN KEY(EmpresaId) REFERENCES Empresas (EmpresaId)  
)
```

```
CREATE TABLE LogOrdens (  
  LogOrdemId INTEGER PRIMARY KEY,  
  OrdemServicoId INTEGER,  
  Data DATETIME,  
  Descricao VARCHAR(150),  
  StatusOrdem INTEGER  
)
```

```
CREATE TABLE CompraProdutos (  
  ProdutoId INTEGER PRIMARY KEY,  
  Quantidade INTEGER,  
  DataCompra DATETIME,  
  StatusCompra INTEGER  
)
```

```
CompraProdutoId INTEGER PRIMARY KEY,  
CompraId INTEGER,  
ProdutoId INTEGER,  
QtdProduto INTEGER,  
Metro DECIMAL(10),  
Unidade INTEGER,  
Valor DECIMAL(10)  
)
```

```
CREATE TABLE Compras (  
CompraId INTEGER PRIMARY KEY,  
EmpresaId INTEGER,  
FornecedorId INTEGER,  
DataCompra DATETIME,  
Total DECIMAL(10),  
FOREIGN KEY(EmpresaId) REFERENCES Empresas (EmpresaId)  
)
```

```
CREATE TABLE Fornecedores (  
FornecedorId INTEGER PRIMARY KEY,  
Endereco VARCHAR(100),  
Telefone VARCHAR(16),  
Fornecedor VARCHAR(150),  
CNPJ VARCHAR(18),  
Ativo INTEGER  
)
```

```
CREATE TABLE Produtos (  
ProdutoId INTEGER PRIMARY KEY,  
PrateleiraId INTEGER,  
Produto VARCHAR(150),  
Ativo INTEGER  
)
```

```
CREATE TABLE Prateleiras (  
PrateleiraId INTEGER PRIMARY KEY,  
CategoriaId INTEGER,  
Descricao VARCHAR(50)  
)
```

```
CREATE TABLE Categorias (  
CategoriaId INTEGER PRIMARY KEY,  
TipoProdutoId INTEGER,  
Categoria VARCHAR(50),  
Ativo INTEGER  
)
```

```
CREATE TABLE TipoProdutos (  
TipoProdutoId INTEGER PRIMARY KEY,  
Tipo VARCHAR(10),  
Ativo INTEGER  
)
```

```
CREATE TABLE OrdemServicos (  
OrdemServicoId INTEGER PRIMARY KEY,  
TipoOrdemServicoId INTEGER,  
UsuarioId INTEGER,  
Descricao VARCHAR(250),  
Cliente VARCHAR(150),  
Status INTEGER,  
FOREIGN KEY(TipoOrdemServicoId) REFERENCES TipoOrdemServicos  
(TipoOrdemServicoId),  
FOREIGN KEY(UsuarioId) REFERENCES Usuarios (UsuarioId)  
)
```

```
ALTER TABLE ProdutosOrdens ADD FOREIGN KEY(OrdemServicoId) REFERENCES  
OrdemServicos (OrdemServicoId)
```

```
ALTER TABLE ProdutosOrdens ADD FOREIGN KEY(ProdutoId) REFERENCES Produtos
(ProdutoId)
ALTER TABLE UsuarioPermissoes ADD FOREIGN KEY(TipoUsuarioId) REFERENCES
TipoUsuarios (TipoUsuarioId)
ALTER TABLE LogOrdens ADD FOREIGN KEY(OrdemServicoId) REFERENCES OrdemServicos
(OrdemServicoId)
ALTER TABLE CompraProdutos ADD FOREIGN KEY(CompraId) REFERENCES Compras
(CompraId)
ALTER TABLE CompraProdutos ADD FOREIGN KEY(ProdutoId) REFERENCES Produtos
(ProdutoId)
ALTER TABLE Compras ADD FOREIGN KEY(FornecedorId) REFERENCES Fornecedores
(FornecedorId)
ALTER TABLE Produtos ADD FOREIGN KEY(PrateleiraId) REFERENCES Prateleiras
(PrateleiraId)
ALTER TABLE Prateleiras ADD FOREIGN KEY(CategoriaId) REFERENCES Categorias
(CategoriaId)
ALTER TABLE Categorias ADD FOREIGN KEY(TipoProdutoId) REFERENCES TipoProdutos
(TipoProdutoId)
```

Tecnologias utilizadas

Documentação:

- StarUml: Para criação dos diagramas de Caso de uso, de Atividades e de Entidade e relacionamento;
- BrModelo: Para criação dos modelos Conceitual, Lógico e Físico do banco de dados.

Prototipo e telas:

- Adobe XD: Software da Adobe de criação de telas de mockups.

Front-end:

- jQuery: Biblioteca de extensão para javascript;
- Bootstrap: Biblioteca que auxilia a estilizar o html;
- PNotify: Biblioteca para criação de notificações da aplicação;
- Alertify: Biblioteca que possui funções semelhantes ao alert() e confirm() do javascript, mas tendo como diferencial o estilo que utiliza bootstrap para tornar mais elegante para a aplicação;
- Datatables: Biblioteca que adiciona funções a tabelas, como campo de pesquisa, ordenação e paginação;
- Highcharts: Biblioteca robusta para criação de gráficos dos mais variados tipos;
- JQuery-mask-plugin: Biblioteca usada para adicionar máscaras aos campos dos formulários;
- JQuery-validation: Biblioteca usada para fazer validação dos formulários;
- Orgchart: Biblioteca usada para criar organogramas;
- Select2: Biblioteca que permite customiza os campos de select adicionando uma caixa de pesquisa;
- Bootstrap4-datetimepicker: Biblioteca que permite adicionar um calendário customizado aos campos de data dos formulários;
- Hover.css: Componente css que adicionar classes que efeitos de hover do mouse;
- Webpack-mix: Biblioteca que funciona como um empacotador de dependências front-end, por meio dele é possível gerar apenas um arquivo .css e um arquivo .js contendo todas as bibliotecas e componentes da aplicação;
- Npm: Ferramenta de linha de comando do Node.js que possibilita gerenciar as dependências front-end;

Back-end:

- C#: Linguagem de programação orientada a objetos;
- Asp.Net Core: Framework web para c#;
- Entity Framework Core: ORM (Object-Relational Mapping) usado para mapear e manipular o banco de dados por meio de objetos no c#;
- Sql Server: Banco de dados da Microsoft;

Controle de versão:

- Github Desktop: Ferramenta usada para trabalhar com o repositório do github usando o git que é uma ferramenta para versionamento.
 - Url do repositório: <https://github.com/kevyncarlos/Estoque>