

IFSP-LIFE

Alexandre Ferreira Pereira de Oliveira – PE3008355

Abril de 2023

Sumário

1	Introdução	3
1.1	Escopo do sistema	3
2	Definição dos requisitos	4
2.1	Requisitos funcionais	4
2.1.1	Funções Fundamentais	4
2.1.2	Funções Básicas.....	5
2.1.3	Funções de Saída	6
2.2	Diagrama de Casos de Uso.....	8
3	Modelos de Domínio.....	9
3.1	Diagrama de Classes	9
4	Modelos de Dados.....	10
4.1	Modelo Lógico (Modelo Workbench).....	10
4.2	Modelo Físico (Script do Banco)	11
5	Interface gráfica	16
5.1	Diagrama navegacional	16
5.2	Layout	17
6	Referências	22

1 Introdução

1.1 Escopo do sistema

O sistema IFSP-LIFE será utilizado em uma farmácia com o principal objetivo de realizar a venda dos produtos. O projeto visa desenvolver uma gestão eficaz do sistema.

Durante o atendimento ao cliente, esse escolhe o(s) produto(s) disponível(is) na farmácia e em seguida, o funcionário localiza o valor dos produtos previamente cadastrados no sistema. Quando a venda ao cliente é finalizada, o valor total da venda é calculado. Após a totalização do pedido, o funcionário solicita ao cliente a forma de pagamento. Caso seja cartão, o total da venda é registrado. Em contrapartida, se for dinheiro, o sistema calcula o troco que será devolvido ao cliente.

O sistema também possui funcionalidades para gerenciar convênios e fornecedores. O Administrador pode cadastrar novos convênios e fornecedores e atualizar as informações dos já existentes, como dados de contato e informações de pagamento. O sistema permite que os convênios tenham descontos especiais nos produtos e serviços da farmácia, e que os fornecedores possam enviar pedidos de compra de produtos.

O sistema conta também com recursos para gerar relatórios gerenciais, que permitem ao Administrador analisar o desempenho da farmácia em diversos aspectos, como vendas, estoque, convênios, fornecedores. Esses relatórios são gerados a partir dos dados armazenados no sistema e podem ser acessados por meio de uma interface amigável e intuitiva. Com essas informações em mãos, o Administrador pode tomar decisões estratégicas e implementar melhorias para otimizar o desempenho da farmácia e garantir a satisfação dos clientes.

Por fim, o sistema tem dois níveis de acesso, o Administrador e o Funcionário. O nível de acesso Funcionário tem permissão para atender aos clientes e realizar pedidos no estabelecimento, enquanto o nível de acesso Administrador tem controle total do sistema, incluindo a manutenção dos dados de usuários, convênios, fornecedores e produtos, além de realizar a compra de novos produtos. Em resumo, o IFSP-LIFE é um software completo que visa melhorar a eficiência e a segurança nas redes farmacêuticas.

2 Definição dos requisitos

2.1 Requisitos funcionais

2.1.1 Funções Fundamentais

RF_FF01 – Efetuar Compra	
Descrição	Essa função permite ao administrador incluir um novo pedido para um determinado fornecedor. Quando é efetuado um novo pedido, é gerado informações sobre o pedido dos produtos comprados, como também é realizado a atualização do estoque. Essa função também permite que a compra possa ser alterada.
Itens de Informações	Identificação da Compra - Código numérico com no máximo 4 casas Identificação do Fornecedor - Código numérico com no máximo 4 casas Identificação do Produto – Código numérico com no máximo 4 casas Quantidade do Produto – Valor numérico inteiro Valor da compra - conjunto numérico; Forma de Pagamento – 1 letra – V – Vista, P – Prazo, C – Cartão

RF_FF02 – Efetuar Venda	
Descrição	Essa função permite ao funcionário incluir uma nova venda para um determinado cliente. Quando é efetuada uma nova venda, é gerada informação sobre o pagamento dos produtos comprados, como também é realizada a atualização do estoque. Essa função também permite que a venda possa ser excluída ou alterada.
Itens de Informações	Identificação do Produto – Código numérico com no máximo 9 casas Descrição do Produto – Texto com tamanho máximo de 50 Quantidade do Produto – Valor numérico inteiro Valor da venda - conjunto numérico; Convênio – 1 letra – S – Possui, N – Não Possui Forma de Pagamento – 1 letra – V – Vista, P – Prazo, C – Cartão

RF_FF03 – Gerenciar Caixa	
Descrição	Essa função permite ao funcionário abrir e fechar o caixa.
Itens de Informações	Código do Caixa – dd/mm/aaaa; Código do Usuário – conjunto numérico; Status – texto com tamanho máximo de 1 – A – Aberto, - F – Fechado; Horário de abertura - hh:mm:ss Valor de abertura – conjunto numérico; Total de entradas – conjunto numérico; Horário de fechamento - hh:mm:ss Total de saídas – conjunto numérico; Saldo – conjunto numérico;

2.1.2 Funções Básicas

RF_FB01 – Manter Itens	
Descrição	Essa função permite a consulta dos itens do estabelecimento como, também, alteração e exclusão
Itens de Informações	Código do Item - conjunto numérico; Nome - texto com tamanho máximo de 255; Dosagem – texto com tamanho máximo de 255; Descrição – texto com tamanho máximo de 255; Lote – conjunto numérico; Data de Fabricação – dd/mm/aaaa; Data de Validade – dd/mm/aaaa; Quantidade – conjunto numérico; Valor Unitário – conjunto numérico;

RF_FB02 – Manter Fornecedores	
Descrição	Essa função permite a consulta dos dados dos fornecedores como, também, a inclusão, alteração e exclusão
Itens de Informações	Código Fornecedor – conjunto numérico; Nome – texto com tamanho máximo de 45; CNPJ – texto com tamanho máximo de 20; Telefone – texto com tamanho máximo de 20; E-mail – texto com tamanho máximo de 45; Endereço – texto com tamanho máximo de 45; Cidade – texto com tamanho máximo de 45; CEP – texto com tamanho máximo de 12; UF – texto com tamanho máximo de 20;

RF_FB03 – Manter Funcionários	
Descrição	Essa função permite a consulta dos funcionários do estabelecimento como, também, alteração e exclusão
Itens de Informações	Código Funcionário – conjunto numérico; Nome – texto com tamanho máximo de 255; CPF – conjunto numérico; Nível (Funcionário ou Administrador) – texto com tamanho máximo de 45; Telefone – texto com tamanho máximo de 20; Endereço – texto com tamanho máximo de 45; Cidade – texto com tamanho máximo de 45; CEP – texto com tamanho máximo de 12; UF – texto com tamanho máximo de 20; Salário – conjunto numérico; Login – texto com tamanho máximo de 45; Senha – texto com tamanho máximo de 45;

RF_FB04 – Manter Convênios	
Descrição	Essa função permite a consulta dos dados de um dos convênios, como, também, a inclusão, alteração e exclusão
Itens de Informações	Código do Convênio - conjunto numérico; Nome - texto com tamanho máximo de 255; Email – texto com tamanho máximo de 255; CNPJ – texto com tamanho máximo de 20; Endereço – texto com tamanho máximo de 255; Telefone – texto com tamanho máximo de 255; Desconto (%) - conjunto numérico;

2.1.3 Funções de Saída

RF_FS01 – Gerar Relatório dos Itens	
Descrição	Essa função emite um relatório que permite a visualização de todos Os produtos com o estoque abaixo do mínimo.
Itens de Informações	Código do Item - conjunto numérico; Nome - texto com tamanho máximo de 255; Dosagem – texto com tamanho máximo de 255; Descrição – texto com tamanho máximo de 255; Lote – conjunto numérico; Data de Fabricação – dd/mm/aaaa; Data de Validade – dd/mm/aaaa; Quantidade – conjunto numérico; Valor Unitário – conjunto numérico;

RF_FS02 – Gerar Relatório de Venda	
Descrição	Essa função emite um relatório de todas as vendas feitas dentro de Um determinado período de tempo.
Itens de Informações	Número da nota fiscal - conjunto numérico; Para cada Venda: Código da Venda - conjunto numérico; Valor da venda- conjunto numérico; Valor da compra - conjunto numérico; Quantidade - conjunto numérico; Valor total - conjunto numérico; Data - dd/mm/aaaa;

RF_FS03 – Gerar Relatório de Compra	
Descrição	Essa função emite um relatório de todas as compras feitas dentro de Um determinado período de tempo.
Itens de Informações	Nome do fornecedor – texto com tamanho máximo de 50; Para cada Compra: Código da Compra - conjunto numérico; Valor - conjunto numérico; Valor da compra - conjunto numérico; Marca - texto com tamanho máximo de 50; Descrição - texto com tamanho máximo de 50; Quantidade - conjunto numérico; Data - dd/mm/aaaa;

2.2 Diagrama de Casos de Uso | IMAGEM MERAMENTE ILUSTRATIVA

Na *Linguagem de modelagem unificada (UML)*, o diagrama de caso de uso resume os detalhes dos usuários do seu sistema (também conhecidos como atores) e as interações deles com o sistema. Para criar um, use um conjunto de símbolos e conectores especializados.

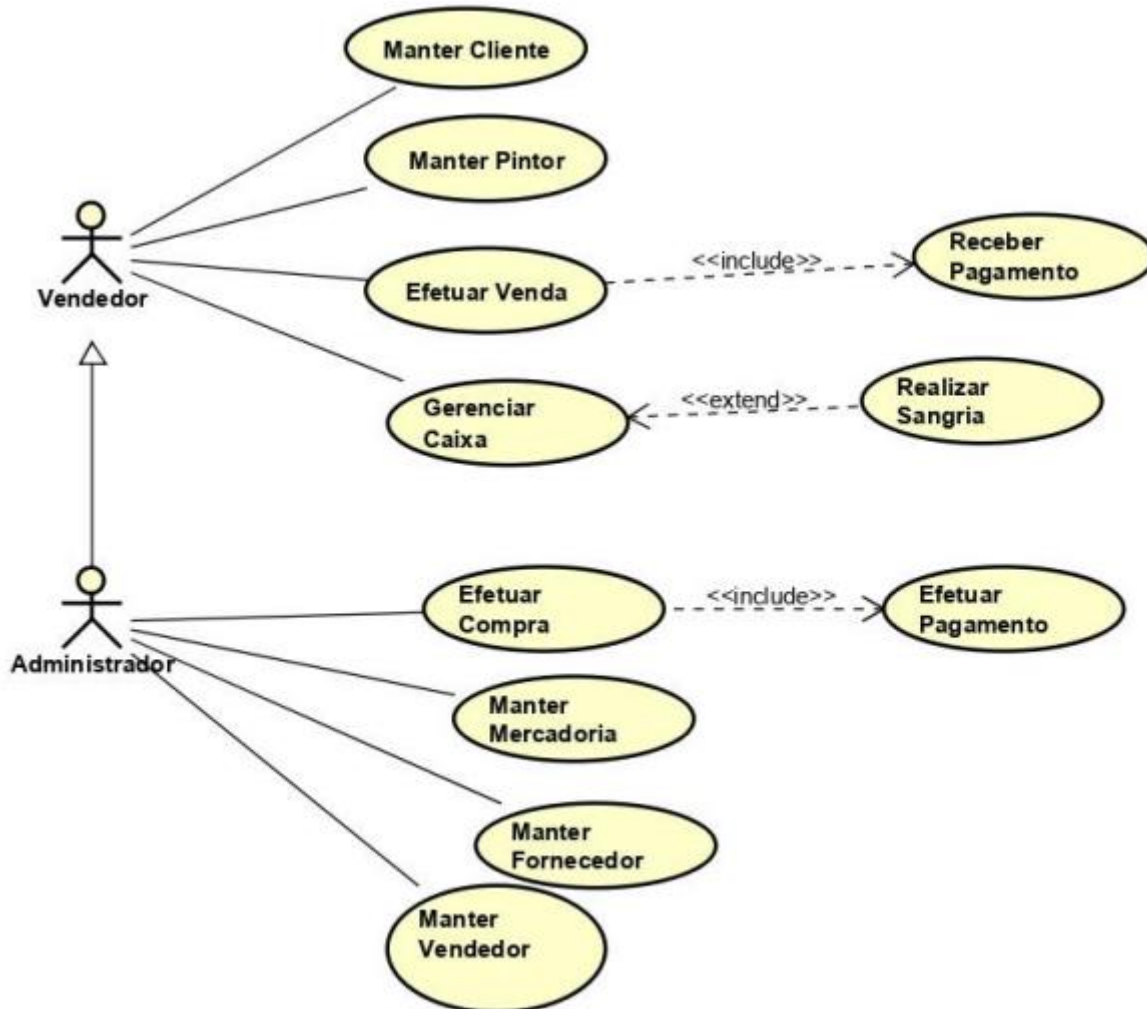


Figura 1: Diagrama de Casos de Uso

3. Modelos de Domínio

3.1 Diagrama de Classes | **IMAGEM MERAMENTE ILUSTRATIVA**

Definição: —Um diagrama de classes mostra um conjunto de classes, interfaces e colaborações e seus relacionamentos. Os diagramas de classes são os diagramas mais encontrados em sistemas de modelagem orientados a objetos. Os diagramas de classes que incluem classes ativas são empregados para direcionar a visão estática do processo de um sistema. (BOOCH; JACOBSON; RUMBAUGH, 2000, p 94).”

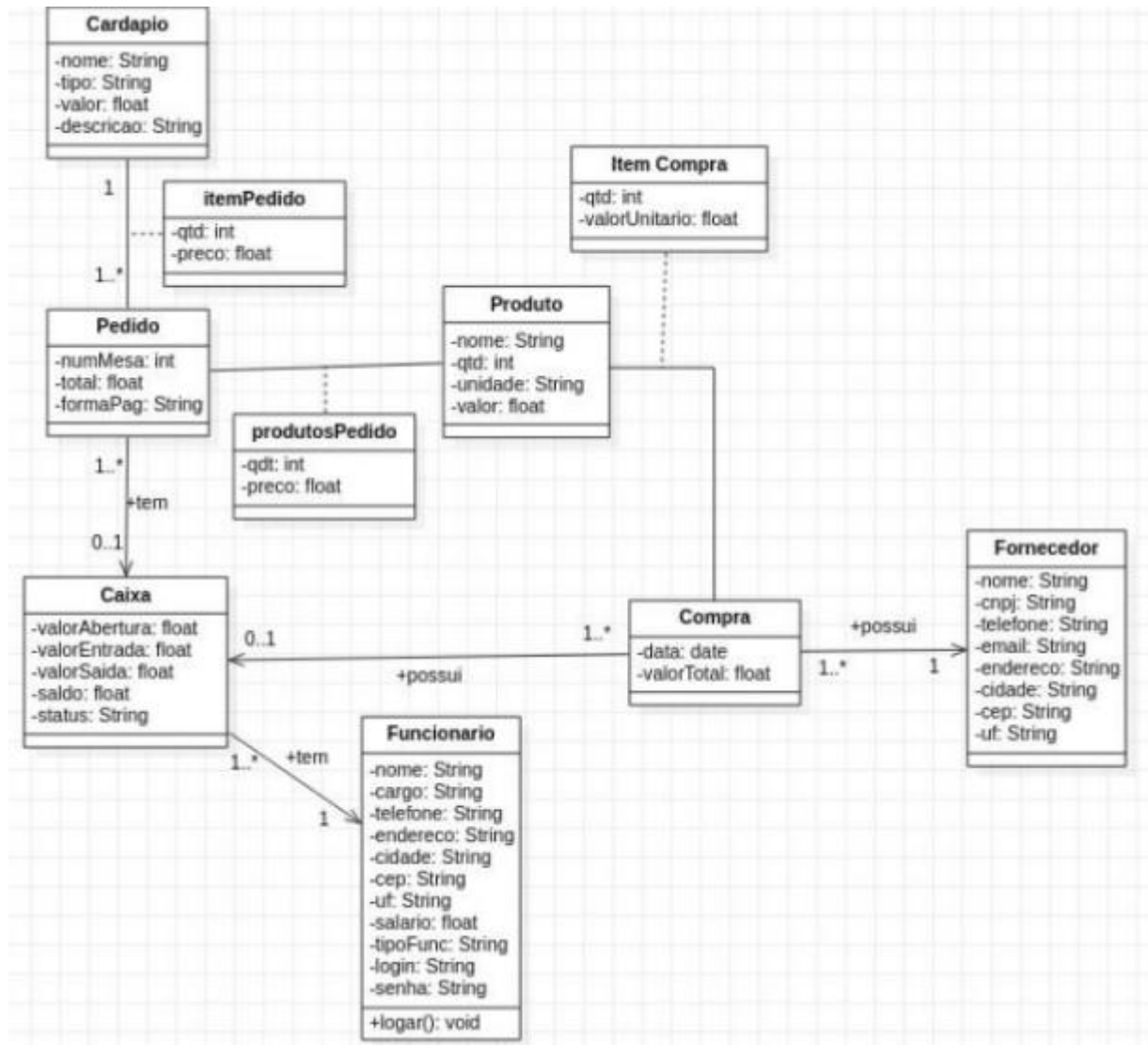


Figura 2: Diagrama de Classe

4. Modelos de Dados

4.1 Modelo Lógico (Modelo Workbench)

Definição: —O Modelo Lógico tem por objetivo representar as estruturas que irão armazenar os dados dentro de um Banco de Dados. Nesse momento é definido a estrutura de seus registros e números de campos com seus respectivos tamanhos

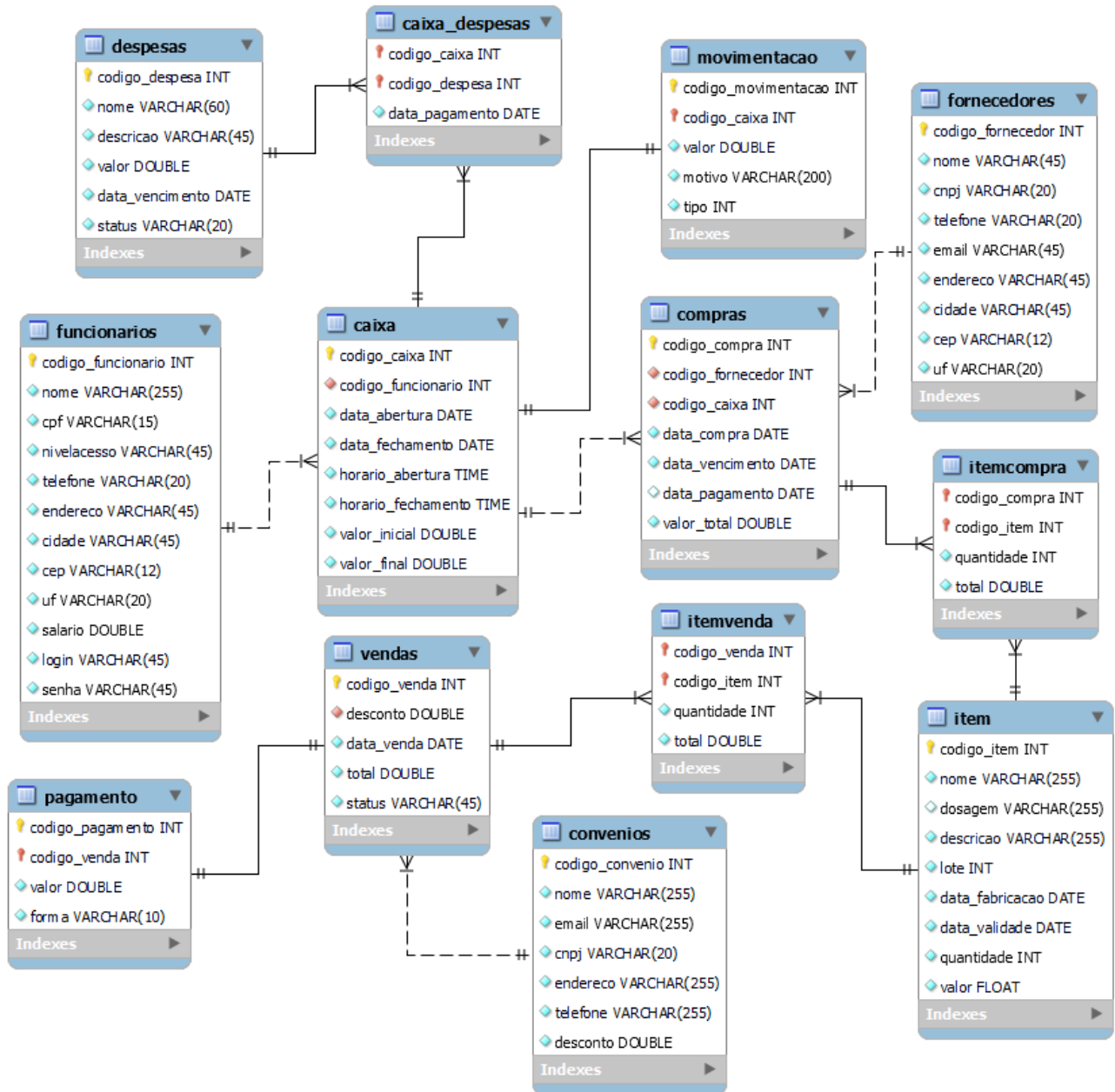


Figura 3: Modelo Lógico

4.2 Modelo Físico (Script do Banco)

Definição: — O Modelo Físico é construído com base em um modelo lógico definido, com intuito de ser aplicado sobre um SGDB. Neste momento entram as questões relacionadas ao tipo e tamanho do campo, relacionamento, indexação, restrições etc. Ele descreve as estruturas físicas de armazenamento, tais como tabelas, índices, gatilhos, funções, visões, nomenclaturas etc.”

```
CREATE DATABASE IF NOT EXISTS ifsplife;  
USE ifsplife;
```

```
CREATE TABLE funcionarios (  
  codigo_funcionario INT NOT NULL AUTO_INCREMENT,  
  nome VARCHAR(255) NOT NULL,  
  cpf VARCHAR(15) NOT NULL,  
  nivelacesso VARCHAR(45) NOT NULL,  
  telefone VARCHAR(20) NOT NULL,  
  endereco VARCHAR(45) NOT NULL,  
  cidade VARCHAR(45) NOT NULL,  
  cep VARCHAR(12) NOT NULL,  
  uf VARCHAR(20) NOT NULL,  
  salario DOUBLE NOT NULL,  
  login VARCHAR(45) NOT NULL,  
  senha VARCHAR(45) NOT NULL,  
  PRIMARY KEY (codigo_funcionario)  
);
```

```
CREATE TABLE caixa (  
  codigo_caixa INT NOT NULL,  
  codigo_funcionario INT NOT NULL,  
  data_abertura DATE NOT NULL,  
  data_fechamento DATE NOT NULL,  
  horario_abertura TIME NOT NULL,  
  horario_fechamento TIME NOT NULL,  
  valor_inicial DOUBLE NOT NULL,  
  valor_final DOUBLE NOT NULL,  
  PRIMARY KEY (codigo_caixa),  
  INDEX fk_caixa_funcionarios1_idx (codigo_funcionario ASC) VISIBLE,  
  CONSTRAINT fk_caixa_funcionarios1  
  FOREIGN KEY (codigo_funcionario)  
  REFERENCES funcionarios (codigo_funcionario)  
);
```

```

CREATE TABLE fornecedores (
codigo_fornecedor INT NOT NULL AUTO_INCREMENT,
nome VARCHAR(45) NOT NULL,
cnpj VARCHAR(20) NOT NULL,
telefone VARCHAR(20) NOT NULL,
email VARCHAR(45) NOT NULL,
endereco VARCHAR(45) NOT NULL,
cidade VARCHAR(45) NOT NULL,
cep VARCHAR(12) NOT NULL,
uf VARCHAR(20) NOT NULL,
PRIMARY KEY (codigo_fornecedor)
);

```

```

CREATE TABLE compras (
codigo_compra INT NOT NULL AUTO_INCREMENT,
codigo_fornecedor INT NOT NULL,
codigo_caixa INT NOT NULL,
data_compra DATE NOT NULL,
data_vencimento DATE NOT NULL,
data_pagamento DATE NULL,
valor_total DOUBLE NOT NULL,
PRIMARY KEY (codigo_compra),
INDEX fk_compras_fornecedores_idx (codigo_fornecedor ASC) VISIBLE,
INDEX fk_compras_caixa1_idx (codigo_caixa ASC) VISIBLE,
CONSTRAINT fk_compras_fornecedores
FOREIGN KEY (codigo_fornecedor)
REFERENCES fornecedores (codigo_fornecedor),
CONSTRAINT fk_compras_caixa1
FOREIGN KEY (codigo_caixa)
REFERENCES caixa (codigo_caixa)
);

```

```

CREATE TABLE convenios (
codigo_convenio INT NOT NULL AUTO_INCREMENT,
nome VARCHAR(255) NOT NULL,
email VARCHAR(255) NOT NULL,
cnpj VARCHAR(20) NOT NULL,
endereco VARCHAR(255) NOT NULL,
telefone VARCHAR(255) NOT NULL,
desconto DOUBLE NOT NULL,
PRIMARY KEY (codigo_convenio)
);

```

```

CREATE TABLE item (
  codigo_item INT NOT NULL AUTO_INCREMENT,
  nome VARCHAR(255) NOT NULL,
  dosagem VARCHAR(255) NULL DEFAULT NULL,
  descricao VARCHAR(255) NOT NULL,
  lote INT NOT NULL,
  data_fabricacao DATE NOT NULL,
  data_validade DATE NOT NULL,
  quantidade INT NOT NULL,
  valor FLOAT NOT NULL,
  PRIMARY KEY (codigo_item)
);

```

```

CREATE TABLE itemcompra (
  codigo_compra INT NOT NULL,
  codigo_item INT NOT NULL,
  quantidade INT NOT NULL,
  total DOUBLE NOT NULL,
  INDEX fk_itemcompra_compras1_idx (codigo_compra ASC) VISIBLE,
  INDEX fk_itemcompra_item1_idx (codigo_item ASC) VISIBLE,
  PRIMARY KEY (codigo_compra, codigo_item),
  CONSTRAINT fk_itemcompra_compras1
    FOREIGN KEY (codigo_compra)
      REFERENCES ifsplife.compras (codigo_compra),
  CONSTRAINT fk_itemcompra_item1
    FOREIGN KEY (codigo_item)
      REFERENCES ifsplife.item (codigo_item)
);

```

```

CREATE TABLE vendas (
  codigo_venda INT NOT NULL AUTO_INCREMENT,
  desconto DOUBLE NOT NULL,
  data_venda DATE NOT NULL,
  total DOUBLE NOT NULL,
  status VARCHAR(45) NOT NULL,
  PRIMARY KEY (codigo_venda),
  INDEX fk_vendas_convenios1_idx (desconto ASC) VISIBLE,
  CONSTRAINT fk_vendas_convenios1
    FOREIGN KEY (desconto)
      REFERENCES ifsplife.convenios (desconto)
);

```

```

CREATE TABLE itemvenda (
  codigo_venda INT NOT NULL,
  codigo_item INT NOT NULL,
  quantidade INT NOT NULL,
  total DOUBLE NOT NULL,
  INDEX fk_itemvenda_vendas1_idx (codigo_venda ASC) VISIBLE,
  INDEX fk_itemvenda_item1_idx (codigo_item ASC) VISIBLE,
  PRIMARY KEY (codigo_venda, codigo_item),
  CONSTRAINT fk_itemvenda_vendas1
    FOREIGN KEY (codigo_venda)
      REFERENCES ifsplife.vendas (codigo_venda),
  CONSTRAINT fk_itemvenda_item1
    FOREIGN KEY (codigo_item)
      REFERENCES ifsplife.item (codigo_item)
);

CREATE TABLE pagamento (
  codigo_pagamento INT NOT NULL,
  codigo_venda INT NOT NULL,
  valor DOUBLE NOT NULL,
  forma VARCHAR(10) NOT NULL,
  PRIMARY KEY (codigo_pagamento, codigo_venda),
  INDEX fk_pagamento_vendas1_idx (codigo_venda ASC) VISIBLE,
  CONSTRAINT fk_pagamento_vendas1
    FOREIGN KEY (codigo_venda)
      REFERENCES ifsplife.vendas (codigo_venda)
);

CREATE TABLE despesas (
  codigo_despesa INT NOT NULL,
  nome VARCHAR(60) NOT NULL,
  descricao VARCHAR(45) NOT NULL,
  valor DOUBLE NOT NULL,
  data_vencimento DATE NOT NULL,
  status VARCHAR(20) NOT NULL,
  PRIMARY KEY (codigo_despesa)
);

CREATE TABLE movimentacao (
  codigo_movimentacao INT NOT NULL,
  codigo_caixa INT NOT NULL,
  valor DOUBLE NOT NULL,
  motivo VARCHAR(200) NOT NULL,
  tipo INT NOT NULL,
  PRIMARY KEY (codigo_movimentacao, codigo_caixa),
  INDEX fk_movimentacao_caixa1_idx (codigo_caixa ASC) VISIBLE,
  CONSTRAINT fk_movimentacao_caixa1
    FOREIGN KEY (codigo_caixa)
      REFERENCES ifsplife.caixa (codigo_caixa)
);

```

```

CREATE TABLE movimentacao (
  `codigo_movimentacao` INT NOT NULL,
  `codigo_caixa` INT NOT NULL,
  `valor` DOUBLE NOT NULL,
  `motivo` VARCHAR(200) NOT NULL,
  `tipo` INT NOT NULL,
  PRIMARY KEY (`codigo_movimentacao`, `codigo_caixa`),
  INDEX `fk_movimentacao_caixa1_idx` (`codigo_caixa` ASC) VISIBLE,
  CONSTRAINT `fk_movimentacao_caixa1`
    FOREIGN KEY (`codigo_caixa`)
      REFERENCES `ifsplife`.`caixa` (`codigo_caixa`)
);

CREATE TABLE caixa_despesas (
  `codigo_caixa` INT NOT NULL,
  `codigo_despesa` INT NOT NULL,
  `data_pagamento` DATE NOT NULL,
  PRIMARY KEY (`codigo_caixa`, `codigo_despesa`),
  INDEX `fk_caixa_has_despesas_despesas1_idx` (`codigo_despesa` ASC) VISIBLE,
  INDEX `fk_caixa_has_despesas_caixa1_idx` (`codigo_caixa` ASC) VISIBLE,
  CONSTRAINT `fk_caixa_has_despesas_caixa1`
    FOREIGN KEY (`codigo_caixa`)
      REFERENCES `ifsplife`.`caixa` (`codigo_caixa`),
  CONSTRAINT `fk_caixa_has_despesas_despesas1`
    FOREIGN KEY (`codigo_despesa`)
      REFERENCES `ifsplife`.`despesas` (`codigo_despesa`)
);

```

5. Interface gráfica

5.1 Diagrama navegacional | **IMAGEM MERAMENTE ILUSTRATIVA**

Definição: —A preocupação do projeto navegacional está relacionada com a forma de navegação do usuário. Esta etapa proporciona uma visão estática do esquema navegacional.”

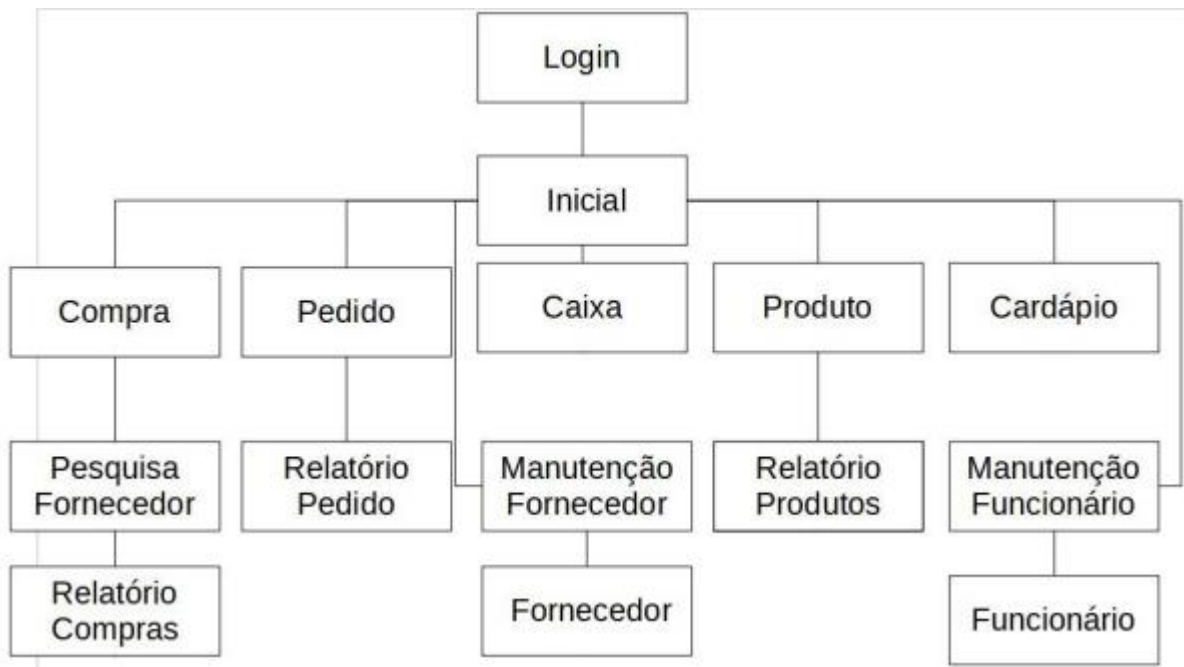


Figura 4: Diagrama Navegacional

5.2 Layout

Definição: —O layout é um esboço ou rascunho que mostra a estrutura física do projeto, ele engloba elementos como texto, gráficos, imagens e a forma como eles se encontram em um determinado espaço.”

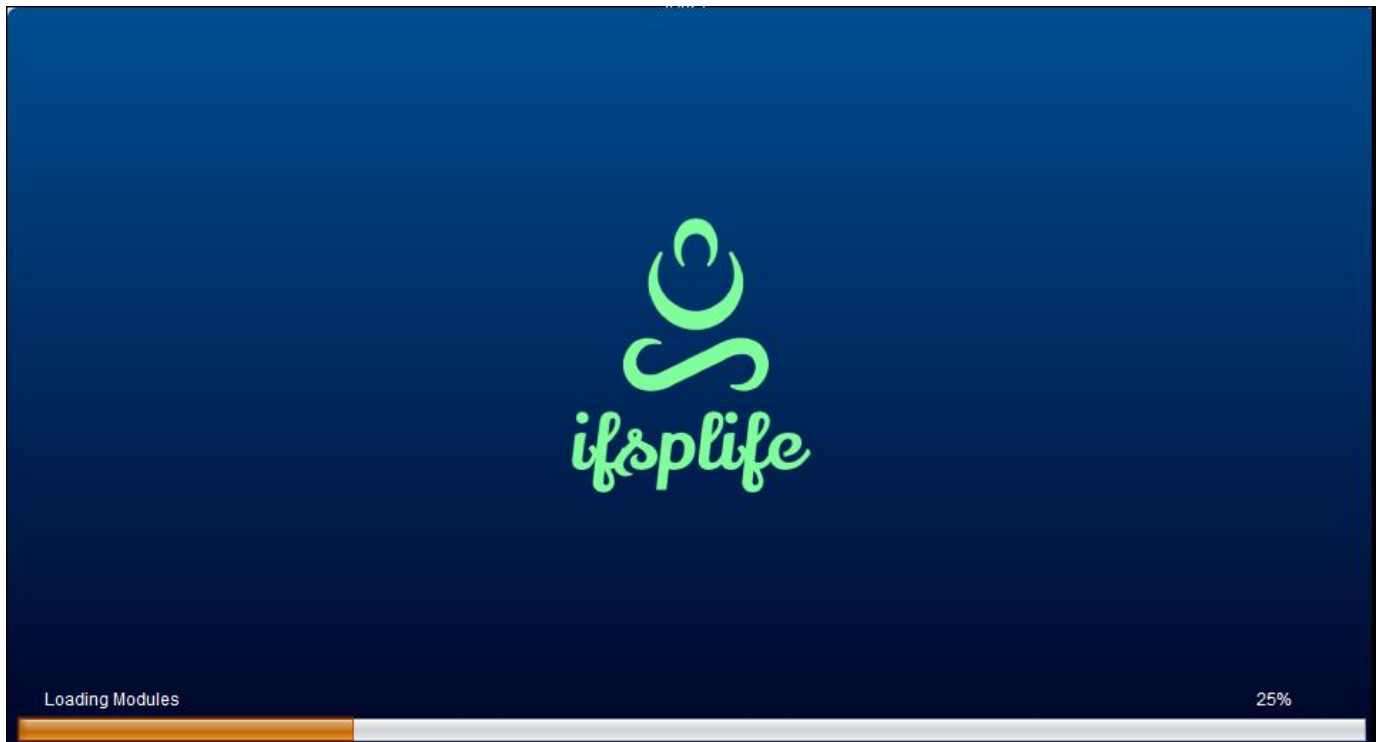


Figura 5: SplashScreen – (Tela de Carregamento)



Figura 6: Tela de Login

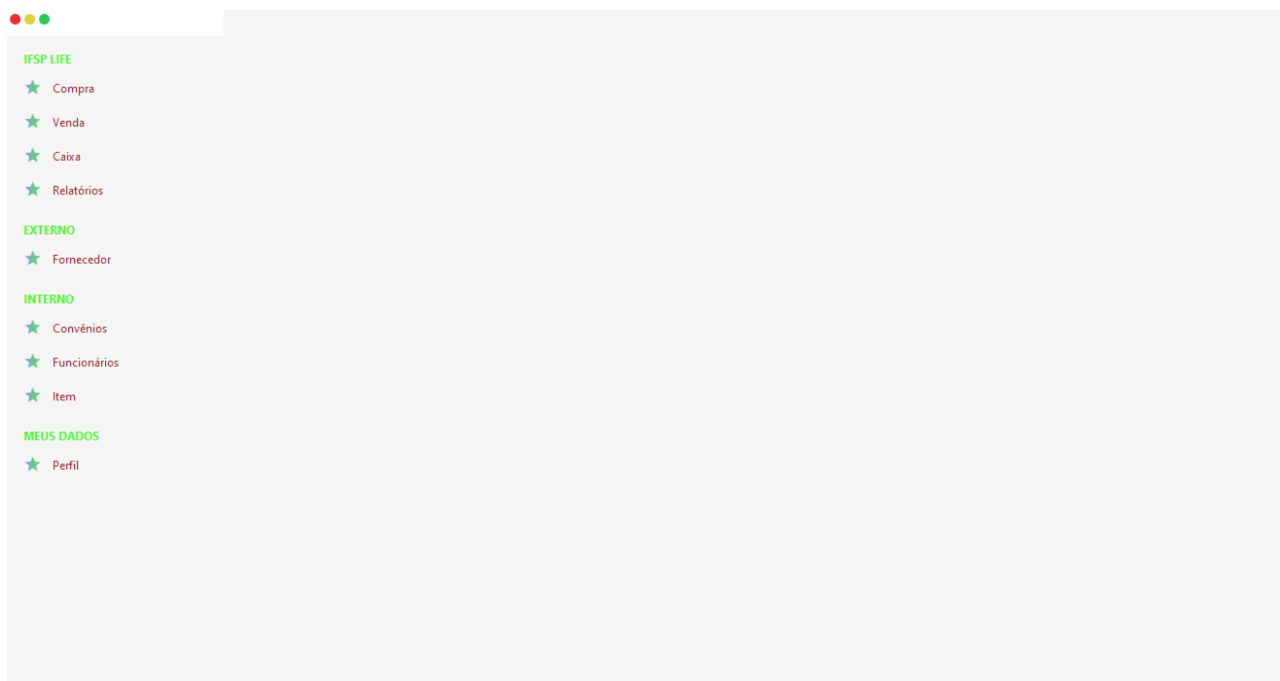


Figura 7: Tela Principal

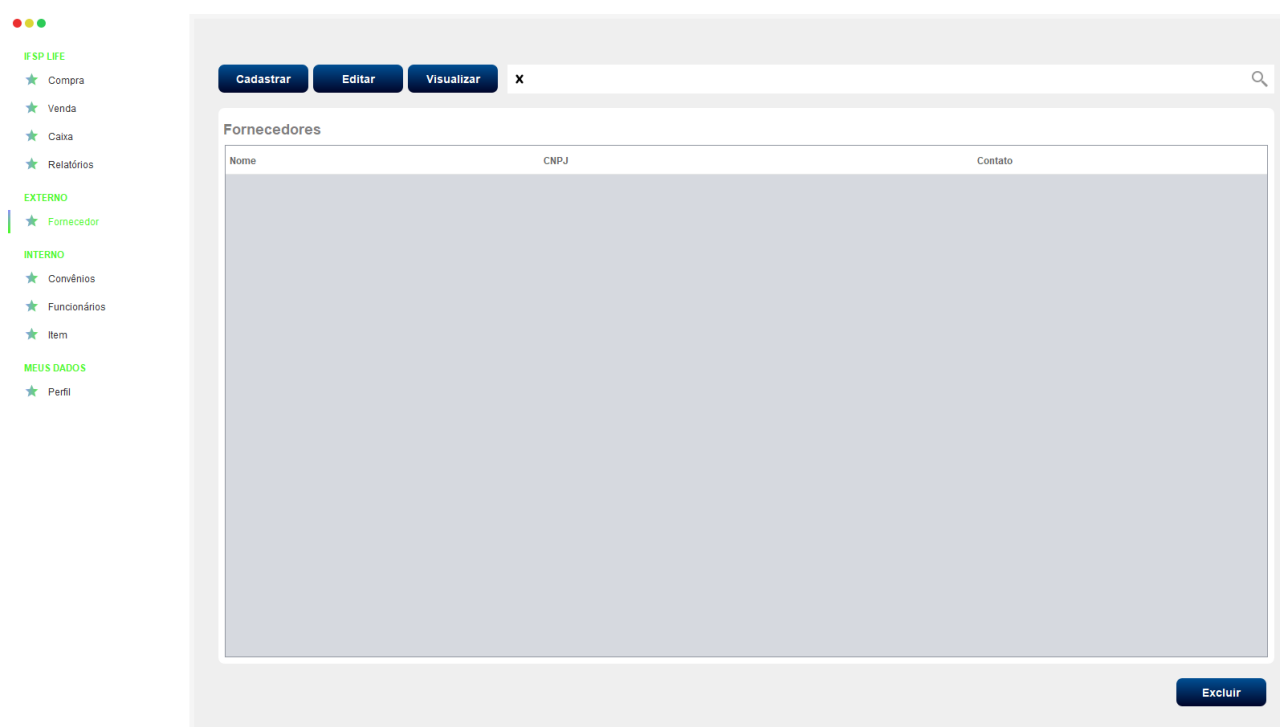


Figura 8: Tela Manutenção de Fornecedores

Cadastro de Fornecedores

Nome 	CNPJ ... / ...	Contato () -	Email
Endereço		Cidade	CEP ... - ...
Estado Acre ▼			

Cadastrar
Cancelar

Figura 9: Tela Dados dos Fornecedores

●
●
●

IFSP LIFE

- ★ Compra
- ★ Venda
- ★ Caixa
- ★ Relatórios

EXTERNO

- ★ Fornecedor

INTERNO

- ★ Convênios
- ★ Funcionários
- ★ Item

MEUS DADOS

- ★ Perfil

Cadastrar
Editar
Visualizar
X

Convênios

Nome	Telefone	Desconto

Excluir

Figura 10: Tela Manutenção de Convênios

Cadastro de Convênios

Nome <input style="width: 90%;" type="text"/>	Email <input style="width: 90%;" type="text"/>	CNPJ <input style="width: 90%;" type="text"/>	Contato <input style="width: 90%;" type="text"/>
Endereco <input style="width: 90%;" type="text"/>	Desconto em [%] <input style="width: 90%;" type="text"/>		

Cadastrar
Cancelar

Figura 11: Tela Dados dos Convênios

●
●
●

IFSP LIFE

- ★ Compra
- ★ Venda
- ★ Caixa
- ★ Relatórios

EXTERNO

- ★ Fornecedor

INTERNO

- ★ Convênios
- ★ Funcionários
- ★ Item

MEUS DADOS

- ★ Perfil

Cadastrar
Editar
Visualizar
x

🔍

Funcionarios

Nome	Email	Cargo

Excluir

Figura 12: Tela Manutenção de Funcionários

Cadastro de Funcionários

Nome	CPF	Cargo	Contato
<input type="text"/>	<input type="text"/>	<input type="text" value="Administrador"/>	<input type="text" value="() -"/>
Endereco	Cidade		CEP
<input type="text"/>	<input type="text"/>		<input type="text" value="-"/>
Estado	Salario	Login	Senha
<input type="text" value="Acre"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Figura 13: Tela Dados dos Funcionários

IFSP LIFE

- ★ Compra
- ★ Venda
- ★ Caixa
- ★ Relatórios
- EXTERNOS
- ★ Fornecedor
- INTERNO
- ★ Convênios
- ★ Funcionários
- ★ Item
- MEUS DADOS
- ★ Perfil

Cadastrar Editar Visualizar X

Item

Nome	Descrição	Desconto

Figura 14: Tela Manutenção de Itens

Cadastro de Itens

Nome

Dosagem

Descricao

Lote

Data de Fabricação

Data de Validade

Quantidade

Valor

Cadastrar

Cancelar

Figura 15: Tela Dados dos Itens

6. Referências