

Projeto Integrador 2º Semestre - DSM

Disciplinas:

Banco de Dados
Desenvolvimento Web II
Engenharia de Software II

Professores:

Bruno
Nilton
Orlando

Grupo(n) / Nome da Empresa: Grupo 6

Sistema: DOT – Sistema de gestão de almoxarife.

Integrante	Papel Principal
João Pedro Pereira dos Santos	PO e Desenvolvedor Back-End
Davi Bonelli Rodrigues	Desenvolvedor Back-End
Davi Samuel Schwartz	Desenvolvedor Front-End
Izabela Kerollayne Pereira Souza	Documentação
Juan Felipe da Silva Santos	Se ausentou

**Fatec Araras
2024**

FICHA DE CONTROLE - PROJETO INTERDISCIPLINAR

DISCIPLINA CHAVE: Engenharia de Software II - PI II
PROFESSOR: Bruno Henrique de Paula Ferreira

GRUPO: Crystal SEMESTRE: 2/2024

TÍTULO DO PROJETO: Gestor de almoxarife.
DATA DA APRESENTAÇÃO: 03/12/2024
NOTA:

INTEGRANTES DO GRUPO: Nome grupo

Nome	Nota Individual
João Pedro Pereira dos Santos	
Davi Bonelli Rodrigues	
Davi Samuel Schwartz	
Izabela Kerollayne Pereira Souza	

Araras, 03 de dezembro de 2024



Fatec Araras "Antônio Brambilla"

Professor Bruno Henrique de Paula Ferreira

Sumário

1.	Apresentação da Empresa.....	5
1.1	Missão	5
1.4	Link Repositório.....	6
2.	Escopo do sistema	6
2.1	Objetivos do projeto	8
2.2	Técnica de levantamento de requisitos	8
2.3	Requisitos funcionais	9
2.4	Requisitos não funcionais	10
2.5	Comparativo entre sistemas.....	10
	Descrever resumidamente e apresentar uma tabela de comparação do sistema desenvolvido versus 1 ou 2 outros sistemas de mercado... Error! Bookmark not defined.	
2.6	Cronograma	11
3.	Documentação do Sistema	12
3.1	Metodologia de Desenvolvimento	12
3.2	Diagramas UML	13
	Diagrama de caso de uso (1º)..... Error! Bookmark not defined.	
	Diagrama de classes (2º)	13
	Diagrama de sequência (2º).....	15
3.3	Modelo Conceitual (1º).....	16
3.4	Modelo lógico (1º)	17
3.5	Modelo Físico (2º)	17
3.6	Recursos e ferramentas (1º e 2º).....	27
3.7	Etapas / Sprints realizados (1º e 2º).....	27
3.8	Interface do usuário.....	27
4.	Testes e Qualidade	35
4.1	Estratégia de Testes: Descrever a estratégia de testes adotada.....	35
4.2	Resultados dos Testes: Apresentar os resultados dos testes realizados	36
4.3	Garantia da Qualidade: Descrever as práticas adotadas para garantia da qualidade	36
4.4	Requisitos mínimos de hardware e software para o sistema	37
4.5	Contrato para desenvolvimento de software	37
5.	Considerações Finais.....	42
6	Referências	42

Anexo I - Diário de bordo	42
Anexo II – Cronograma efetivo	43
Anexo III – Evidências	43

1. Apresentação da Empresa



1.1 Missão

Nosso projeto tem como missão otimizar e realizar gestão de estoque com eficiência, trazendo mais agilidade e priorizando a discrição e segurança dos dados de nossos clientes.

1.2 Visão

Desenvolver produtos software e serviços que alcancem destaque no cenário de softwares de gestão.

1.3 Valores

- Priorizamos a privacidade e total segurança colaboradores
- Trazer eficiência através de nossos produtos
- Estabelecer uma conexão de parceria com nossos clientes
- Constante melhoria de nossos processos
- Criar soluções inovadoras

1.4 Link Repositório

[jp8002/PI_FATEC_2024_2-Semestre: Sistema para controle de almoxarifado desenvolvido em php \(github.com\)](https://github.com/jp8002/PI_FATEC_2024_2-Semestre_Sistema_para_controle_de_almoxarifado)

2. Escopo do sistema

Nome do Sistema: DOT

Escopo do Sistema

- **Funcionalidades do Almoxarifado:**

I. Cadastro de E.P.I (O usuário utiliza essa função para adicionar um E.P.I e suas informações ao banco de dados).

II. Remover E.P.I do Estoque (O usuário utiliza essa função para remover um E.P.I já existente no banco de dados).

III. Atualizar Estoque (O usuário utiliza essa função para modificar/atualizar algum dado de um E.P.I já existe no bando de dados).

IV. Registrar Saída de E.P.I (O usuário utiliza essa função para registrar uma retirada de E.P.I para uso, salvando dados como: Nome, data, E.P.I, C.A e quantidade, armazenando no banco de dados).

V. Registrar Devolução (O usuário utiliza essa função para registrar uma devolução de E.P.I, salvando dados como: Nome do funcionário, data, E.P.I, C.A e quantidade, armazenando no banco de dados).

VI. Registrar entrada (O usuário utiliza essa função para registrar uma entrada/compra de E.P.I, salvando dados como: Nome do fornecedor, data, E.P.I, C.A e quantidade, armazenando no banco de dados).

VII. Buscar item do Estoque (O usuário utiliza essa função para ver todos os E.P.Is existentes no banco de dados, podendo visualizar a quantidade, C.A, Nome do E.P.I, numeração e data de validade).

- **Funcionalidades do Supervisor:**

- I. **Buscar Histórico de compras** (O usuário utiliza essa função para ver o histórico de compras de E.P.I).

- **Funcionalidades universais:**

I. Buscar Histórico de saída (O usuário utiliza essa função para visualizar todos os dados salvos da função "Registrar Saída" no banco de dados).

II. Buscar Histórico de Entradas (O usuário utiliza essa função para visualizar os dados salvos da função "Registrar Entrada" no banco de dados).

III. Buscar Histórico de Devolução (O usuário utiliza essa função para visualizar os dados salvos da função "Registrar Devolução" no banco de dados).

IV. Buscar Histórico de Devolução (O usuário utiliza essa função para visualizar os dados salvos da função "Registrar Devolução" no banco de dados).

Características do Sistema:

- **Interface:**
 - Design intuitivo e funcional que facilita a navegação do usuário.
 - Funções personalizadas para cada usuário.
 - Paleta de cores pensada para trazer tranquilidade ao usuário.
- **Integração:**
 - Sistema pensado para realizar uma gestão de almoxarifado com eficiência contendo alertas de baixa de estoque.
 - Facilita a interação entre usuários.
 - Relatórios de estoque.
 - Históricos de entrada e saída.

2.1 Objetivos do projeto

Desenvolver um sistema de controle de estoque de almoxarifados. Com isso pretendemos criar uma alternativa mais eficiente do que o uso de meios físicos para registrar as movimentações de estoque.

2.2 Técnica de levantamento de requisitos

- Observação do cenário: Um dos integrantes do nosso PI, Davi Samuel Schwartz, trabalha no setor que utilizará o sistema a ser desenvolvido, com isso ele passou a observar como é o fluxo de trabalho na área de almoxarifado.
- Entrevista com stakeholder

2.3 Requisitos funcionais

Nº	Prioridade	Descrição
RF1	Essencial	CADASTRAR ALMOXARIFE - O sistema deve permitir que o usuário realize o cadastro de um Almoхарife ao sistema
RF2	Essencial	LOGIN - O Sistema deve permitir que o usuário com o cadastro realizado, possa realizar o "Login" e ter acesso as funcionalidades de acordo com sua função.
RF3	Essencial	CADASTRAR E.P.I - O Sistema deve permitir ao usuário cadastrado e com o "Login" efetuado poder cadastrar um novo E.P.I ao estoque.
RF4	Não essencial	REMOVER E.P.I - O sistema deve permitir que o usuário cadastrado e com o "Login" efetuado pode remover um E.P.I já cadastrado no estoque.
RF5	Essencial	ATUALIZAR ESTOQUE - O sistema deve permitir que o usuário cadastrado e com o "Login" efetuado pode atualizar as informações de um E.P.I já cadastrado no estoque.
RF6	Essencial	REGISTRAR SAÍDA - O sistema deve permitir que o usuário cadastrado e com o "Login" efetuado pode registrar a saída de um ou vários E.P.I já cadastrado no estoque.
RF7	Essencial	REGISTRAR COMPRA - O sistema deve permitir que o usuário cadastrado e com o "Login" efetuado pode registrar a compra de um ou vários E.P.I já cadastrado no estoque.
RF8	Essencial	REGISTRAR DEVOLUÇÃO - O sistema deve permitir que o usuário cadastrado e com o "Login" efetuado pode registrar a devolução de um ou vários E.P.I que já foram retirados do estoque.
RF9	Essencial	BUSCAR ITEM DO ESTOQUE - O sistema deve permitir que o usuário cadastrado e com o "Login" efetuado pode visualizar todos os itens existentes no estoque e pesquisar um item específico.
RF10	Essencial	BUSCAR HISTÓRICO DE SAÍDAS - O sistema deve permitir que o usuário cadastrado e com o "Login" efetuado possa visualizar todos os registros de saídas de E.P.I.
RF11	Essencial	BUSCAR HISTÓRICO DE COMPRAS - O sistema deve permitir que o usuário cadastrado e com o "Login" efetuado possa visualizar todos os registros de compras de E.P.I.
RF12	Essencial	BUSCAR HISTÓRICO DE DEVOLUÇÃO - O sistema deve permitir que o usuário cadastrado e com o "Login" efetuado possa visualizar todos os registros de devolução de E.P.I.
RF13	Essencial	BUSCAR HISTÓRICO GERAL - O sistema deve permitir que o usuário cadastrado e com o "Login" efetuado possa visualizar todos os registros de Saídas, Compras e Devoluções de E.P.I.
RF14	Essencial	CADASTRAR FUNCIONÁRIO - O sistema deve permitir que o usuário cadastrado e com o "Login" efetuado possa cadastrar um funcionário que poderá retirar ou devolver um E.P.I do estoque.
RF15	Essencial	CADASTRAR FORNECEDOR - O sistema deve permitir que o usuário cadastrado e com o "Login" efetuado possa cadastrar um fornecedor de E.P.I.

2.4 Requisitos não funcionais

Nº	Prioridade	Descrição
RN1	Obrigatório	SISTEMA RÁPIDO - O sistema deve ser capaz de processar consultas e exibir resultados em no máximo 3 segundos.
RN2	Obrigatório	SEGURANÇA - O acesso ao sistema deve ser autenticado, com suporte a autenticação e multifator para administradores.
RN3	Obrigatório	ESCALABILIDADE - O sistema deve suportar o aumento do número de usuários simultâneos sem degradação perceptível no desempenho.
RN4	Obrigatório	MANUTENIBILIDADE - O código do sistema deve ser modular e seguir padrões de codificação para facilitar a manutenção.
RN5	Obrigatório	SERVIDOR SQL - A máquina onde o sistema será utilizado deve possuir servidores com suporte para SQL e PHP (xampp por exemplo).

2.5 Comparativo entre sistemas

Tabela 1 – Comparativo das funcionalidades da aplicação

TOTVS:

- **Funcionalidades:** Sistema robusto, com módulos para todas as áreas de gestão (financeiro, RH, estoque etc.), além de soluções verticais para setores específicos (indústria, saúde, agronegócio).
- **Usabilidade:** Complexo, com maior curva de aprendizado e necessidade de treinamento.
- **Customização:** Alta flexibilidade e personalização, mas com custos e tempo de implementação elevados.
- **Custo:** Mais caro, tanto na aquisição quanto na implementação e manutenção.
- **Suporte:** Bem estruturado, com opções de atendimento 24/7 e consultoria.
- **Tecnologia:** Inovações em IA, automação, cloud computing.

Logix:

- **Funcionalidades:** ERP mais simples, com módulos essenciais (financeiro, estoque, vendas etc.).
- **Usabilidade:** Intuitivo e fácil de usar, com rápida implementação.
- **Customização:** Menos flexível, mais focado em soluções prontas e práticas.
- **Custo:** Mais acessível e adequado para empresas com orçamentos menores.
- **Suporte:** Atendimento personalizado, mas com alcance e escopo menores que o TOTVS.

Fatec Araras "Antônio Brambilla"

- **Tecnologia:** Focado em automação de processos, com menor ênfase em inovações complexas.

DOT:

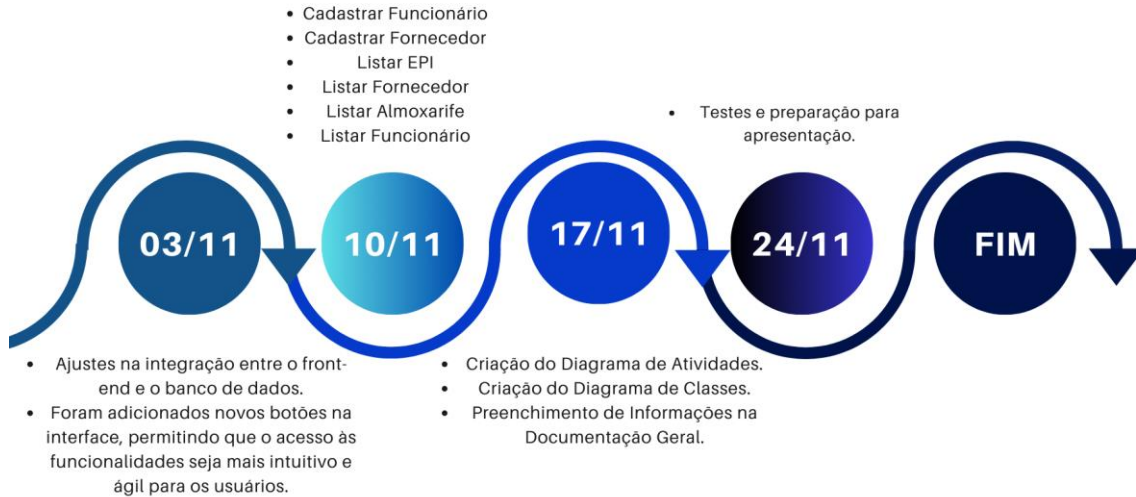
- **Funcionalidades:** Sistema leve que facilita o acesso mesmo em dispositivos mais antigos, desde que use do Windows 7 em diante.
- **Usabilidade:** Pensado para ser intuitivo facilitando a acessibilidade para operadores de terceira idade.
- **Customização:** Flexível, com custos compatíveis com a complexidade da implementação.
- **Custo:** Acessível, adequado para empresas que pretende realizar um investimento único por algo vitalício tendo apenas as atualizações periódicas e implementação caso o cliente desejar.
- **Suporte:** Suporte diretamente com os desenvolvedores que atenderá de forma personalizada.
- **Tecnologia:**

2.6 Cronograma



SPRINT

Progresso semanal

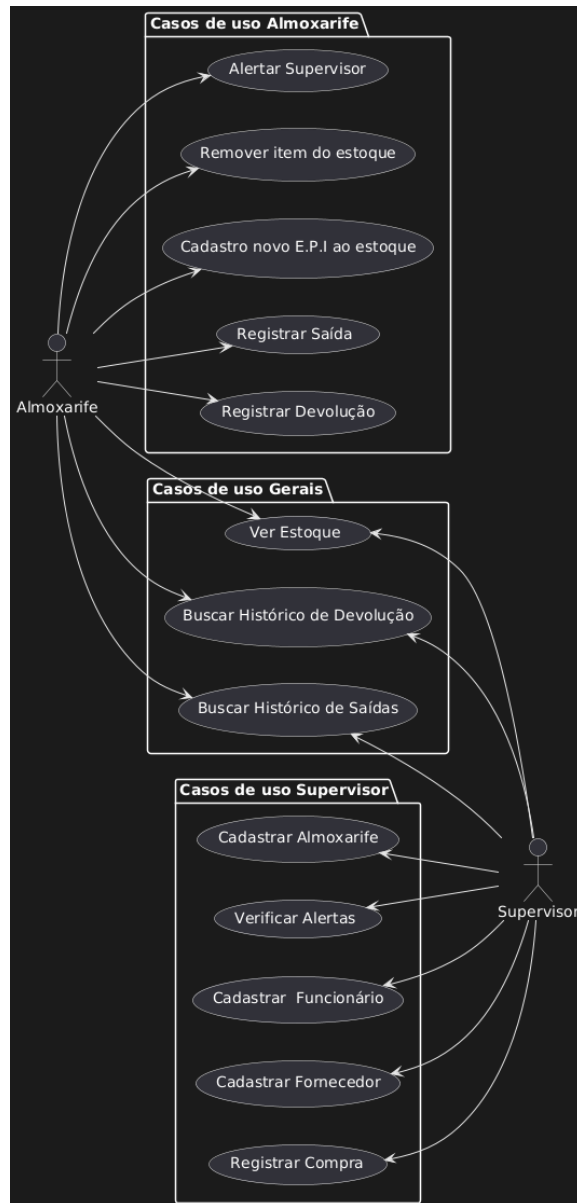


3. Documentação do Sistema

3.1 Metodologia de Desenvolvimento

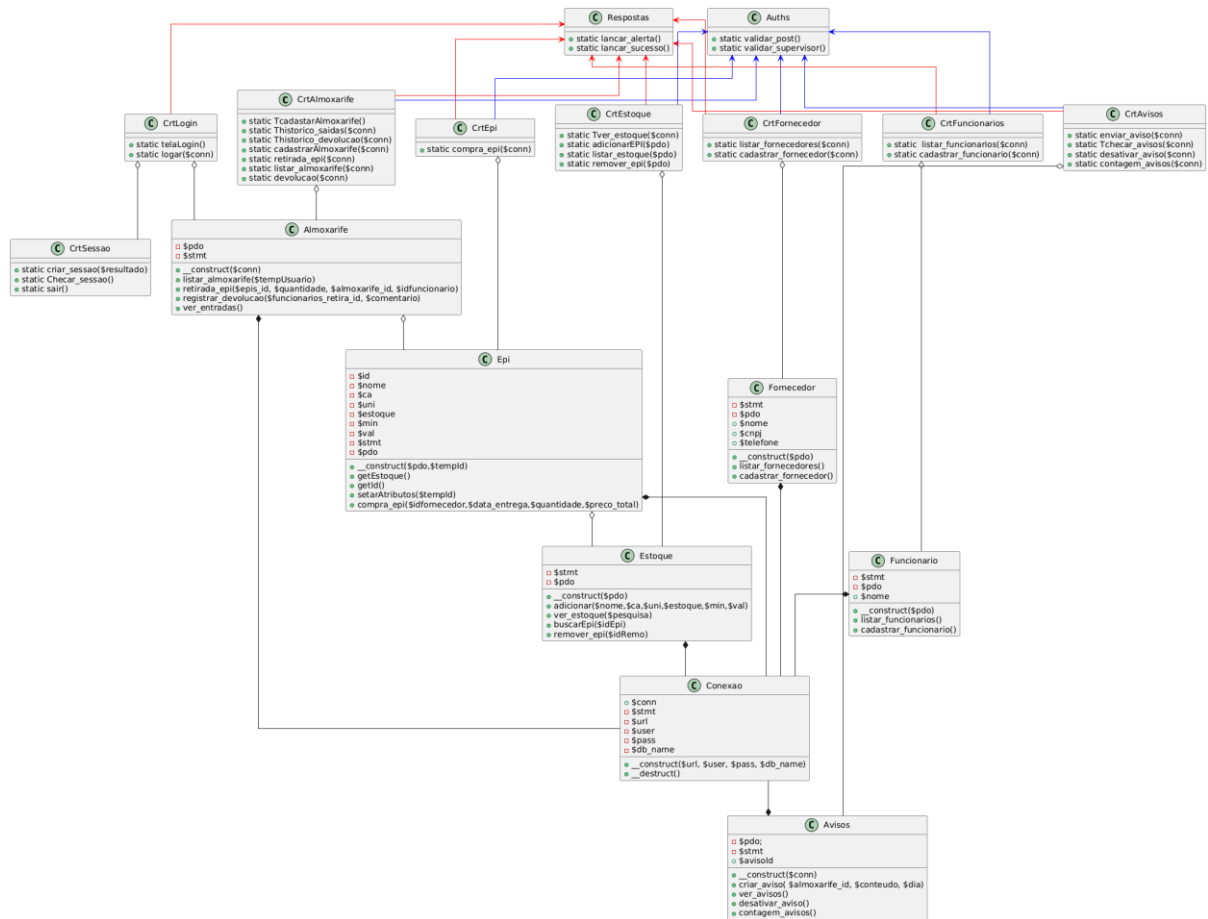
O SCRUM foi utilizado como a metodologia de desenvolvimento tendo espaços curtos para cada reunião com entregas pontuais gerando uma constância de sprints. Foi a metodologia escolhida pois é dinâmico, flexível e centrado na entrega de valor contínuo que possibilitou mais que desenvolver um software, mas nos relacionarmos com nosso cliente e construir um produto juntos a fim de resolver um problema.

3.2 Diagramas UML: Diagrama de caso de uso (1º)

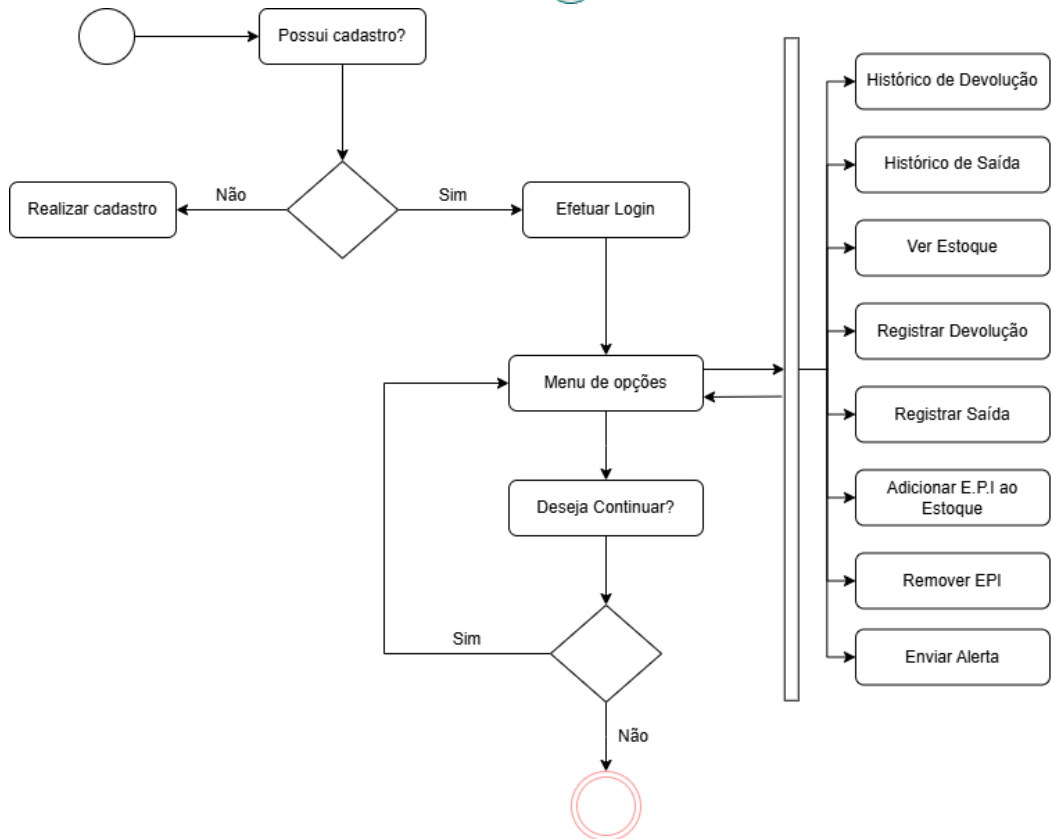
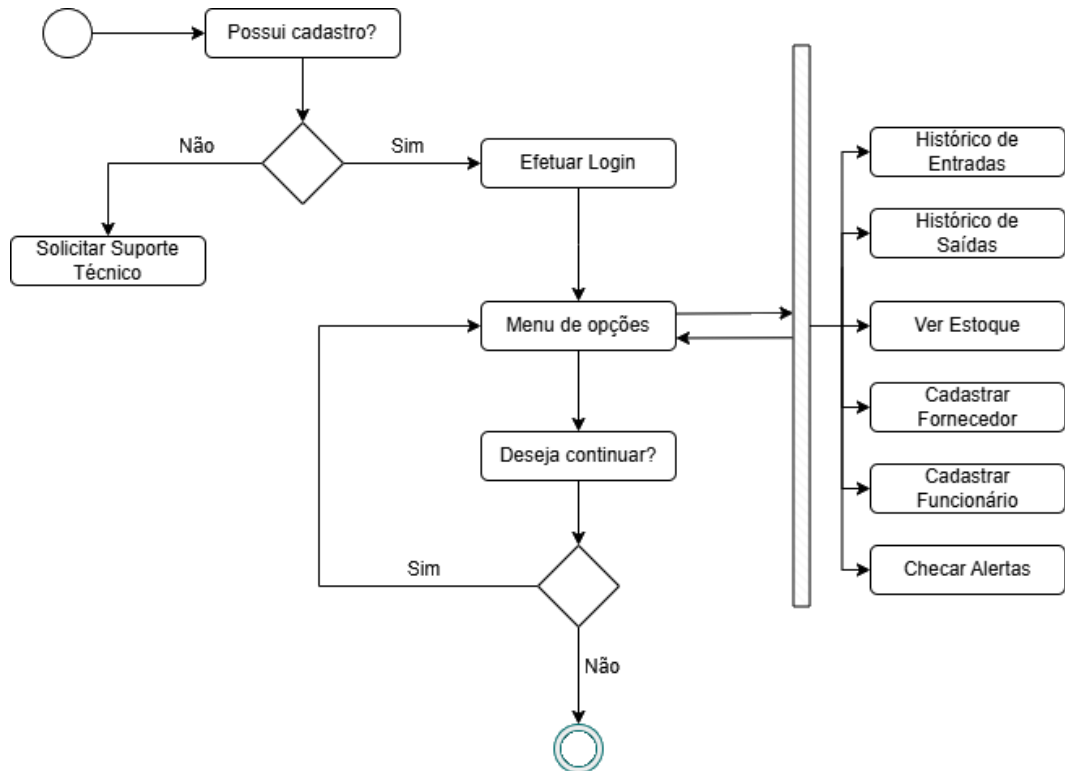


3.3 Diagrama de classes (2º)

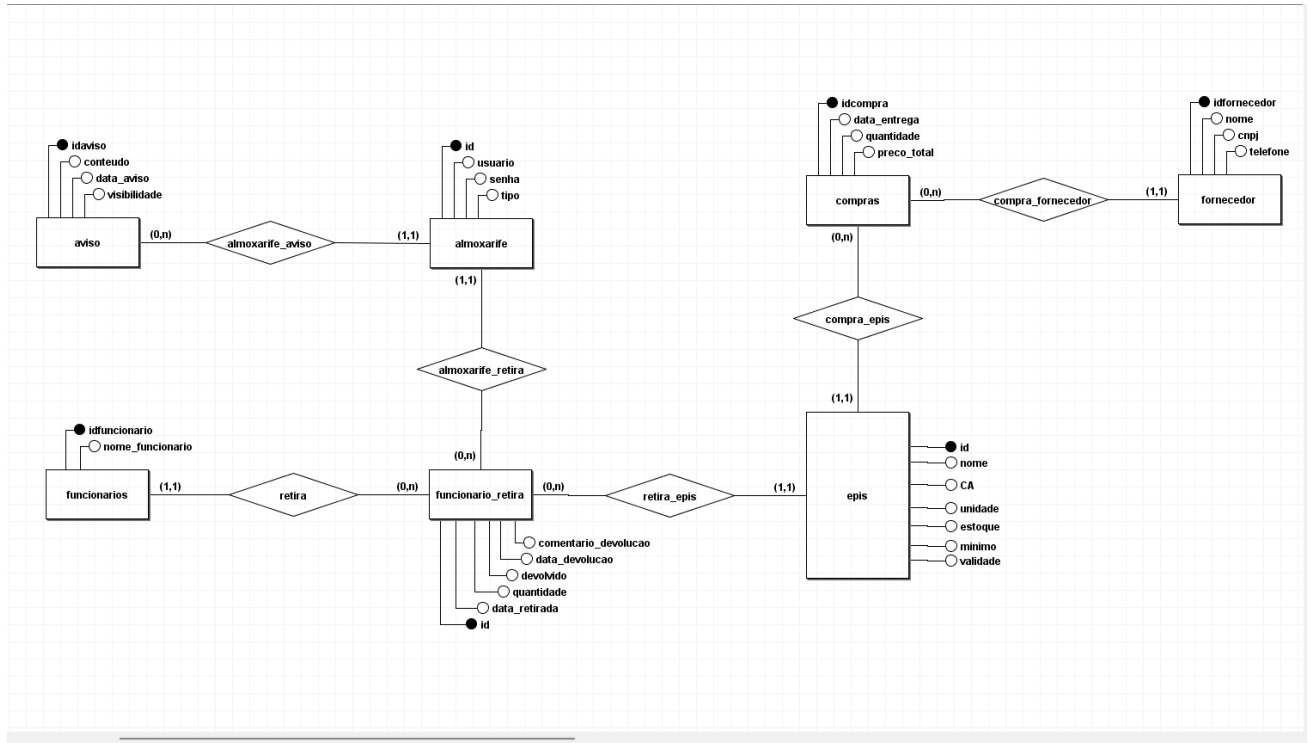
Fatec Araras "Antônio Brambilla"



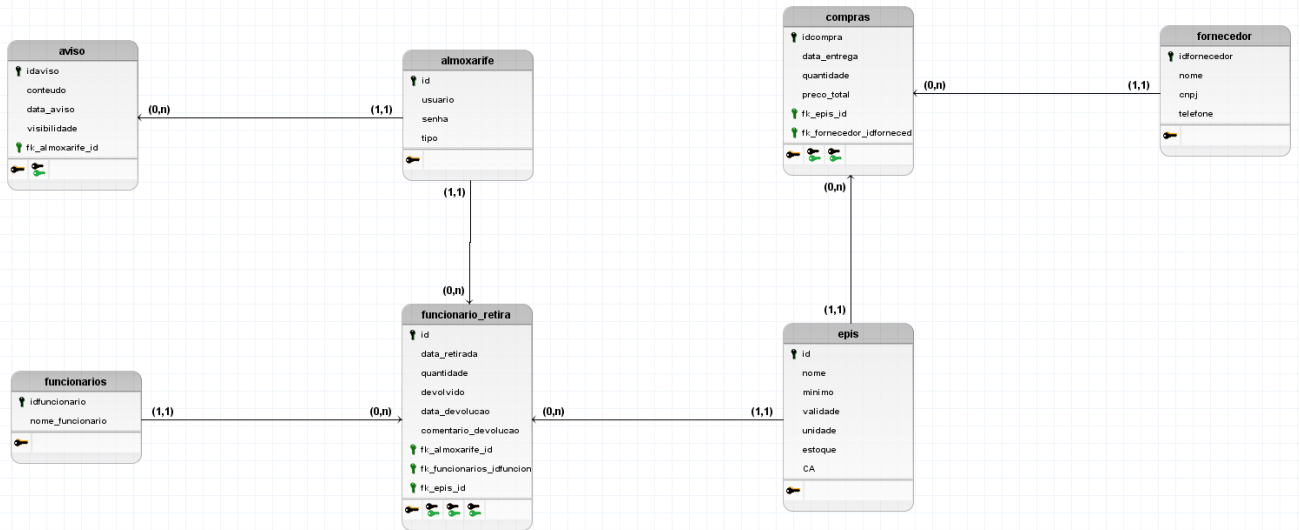
3.4 Diagrama de Atividade



3.4 Modelo Conceitual (1º)



3.5 Modelo lógico (1º)



3.6 Modelo Físico (2º)

```

-- phpMyAdmin SQL Dump
-- version 5.2.1
-- https://www.phpmyadmin.net/
--
-- Host: 127.0.0.1
-- Generation Time: Nov 26, 2024 at 06:29 PM
-- Server version: 10.4.32-MariaDB
-- PHP Version: 8.0.30
    
```

```

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
START TRANSACTION;
SET time_zone = "+00:00";
    
```

```

/*!40101 SET
@OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET
@OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
    
```

```
/*!40101 SET
@OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;

--
-- Database: `almoxarifado`
--
CREATE DATABASE IF NOT EXISTS `almoxarifado` DEFAULT CHARACTER SET
utf8mb4 COLLATE utf8mb4_general_ci;
USE `almoxarifado`;

DELIMITER $$
--
-- Procedures
--
DROP PROCEDURE IF EXISTS `atualiza_estoque`$$
CREATE DEFINER=`root`@`localhost` PROCEDURE `atualiza_estoque` (IN `id_epi`
INT, IN `quantidade` INT, IN `acao` CHAR) begin

if acao = "R" then
UPDATE epis SET estoque = estoque - quantidade WHERE id = id_epi;

ELSEif acao = "D" then
UPDATE epis SET estoque = estoque + quantidade WHERE id = id_epi;
end if;
END$$

DROP PROCEDURE IF EXISTS `cadastrar_fornecedor`$$
CREATE DEFINER=`root`@`localhost` PROCEDURE `cadastrar_fornecedor` (IN
`nome` CHAR(100), IN `cnpj` CHAR(14), `telefone` CHAR(11)) BEGIN

INSERT INTO fornecedor VALUES(null, nome, cnpj, telefone);

END$$

DROP PROCEDURE IF EXISTS `cadastrar_funcionario`$$
CREATE DEFINER=`root`@`localhost` PROCEDURE `cadastrar_funcionario` (IN
`nome` CHAR(25)) BEGIN

INSERT INTO funcionarios VALUES(null, nome);
```

END\$\$

```
DROP PROCEDURE IF EXISTS `registrar_devolucao`$$
CREATE DEFINER=`root`@`localhost` PROCEDURE `registrar_devolucao` (IN
`id_retirada` INT, IN `comentario` CHAR(255)) begin
```

```
    UPDATE funcionarios_retira SET devolvido = 1, data_devolucao = NOW(),
comentario_devolucao = comentario WHERE id = id_retirada;
```

END\$\$

```
DROP PROCEDURE IF EXISTS `registrar_saida`$$
CREATE DEFINER=`root`@`localhost` PROCEDURE `registrar_saida` (IN `idepi`
INT, IN `idalmoxarife` INT, IN `nquantidade` INT, IN `id_funcionario` INT) BEGIN
```

```
INSERT INTO funcionarios_retira (epis_id, almoxarife_id, data_retirada, quantidade,
funcionarios_idfuncionario)
VALUES (idepi, idalmoxarife, NOW(), nquantidade, id_funcionario);
```

END\$\$

```
DROP PROCEDURE IF EXISTS `ver_entradas`$$
CREATE DEFINER=`root`@`localhost` PROCEDURE `ver_entradas` () BEGIN
```

```
SELECT fr.id, f.nome_funcionario, e.nome, fr.data_retirada, fr.data_devolucao,
fr.comentario_devolucao
FROM funcionarios_retira fr, epis e, funcionarios f
where fr.funcionarios_idfuncionario = f.idfuncionario and fr.epis_id = e.id and
fr.devolvido = 1;
```

END\$\$

```
DROP PROCEDURE IF EXISTS `ver_estoque`$$
CREATE DEFINER=`root`@`localhost` PROCEDURE `ver_estoque` (IN `pesquisa`
CHAR(45)) BEGIN
```

```
SELECT * FROM epis e WHERE e.nome like pesquisa;
```

END\$\$

```
DROP PROCEDURE IF EXISTS `ver_saidas`$$
```

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `ver_saidas` () BEGIN

SELECT fr.id, e.nome, f.nome_funcionario, fr.quantidade, fr.data_retirada, a.usuario
FROM funcionarios_retira fr, almoxarife a, epis e, funcionarios f
WHERE fr.epis_id = e.id and fr.almoxarife_id = a.id and fr.funcionarios_idfuncionario
= f.idfuncionario;

END$$

DELIMITER ;

-- -----

--
-- Table structure for table `almoxarife`
--

DROP TABLE IF EXISTS `almoxarife`;
CREATE TABLE `almoxarife` (
  `id` int(10) UNSIGNED NOT NULL,
  `usuario` varchar(45) NOT NULL,
  `senha` varchar(60) NOT NULL,
  `tipo` varchar(25) DEFAULT 'normal'
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

--
-- Dumping data for table `almoxarife`
--

INSERT INTO `almoxarife` (`id`, `usuario`, `senha`, `tipo`) VALUES
(1, 'adm', '$2a$12$5ws2ttzW4bk8.W11ZS5JzefOiZ4YHUV.CE2dsqJ32QcOCEp9Ka6Zu', 'supervisor');

-- -----

--
-- Table structure for table `aviso`
--

DROP TABLE IF EXISTS `aviso`;
```

```
CREATE TABLE `aviso` (  
  `idaviso` int(10) UNSIGNED NOT NULL,  
  `almoxarife_id` int(10) UNSIGNED NOT NULL,  
  `conteudo` text NOT NULL,  
  `data_aviso` date NOT NULL,  
  `visibilidade` tinyint(1) DEFAULT 1  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
```

--

-- Table structure for table `compras`

--

```
DROP TABLE IF EXISTS `compras`;  
CREATE TABLE `compras` (  
  `idcompra` int(10) UNSIGNED NOT NULL,  
  `epis_id` int(10) UNSIGNED NOT NULL,  
  `fornecedor_idfornecedor` int(10) UNSIGNED NOT NULL,  
  `data_entrega` date NOT NULL,  
  `quantidade` int(10) UNSIGNED NOT NULL,  
  `preco_total` float NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
```

--

-- Triggers `compras`

--

```
DROP TRIGGER IF EXISTS `TGR_compra_epi`;  
DELIMITER $$  
CREATE TRIGGER `TGR_compra_epi` AFTER INSERT ON `compras` FOR EACH  
ROW begin
```

```
UPDATE epis e SET estoque = estoque + NEW.quantidade WHERE NEW.epis_id=  
e.id;
```

```
END
```

```
$$
```

```
DELIMITER ;
```

```
--  
-- Table structure for table `epis`  
--
```

```
DROP TABLE IF EXISTS `epis`;  
CREATE TABLE `epis` (  
  `id` int(10) UNSIGNED NOT NULL,  
  `nome` varchar(45) NOT NULL,  
  `CA` varchar(20) NOT NULL,  
  `unidade` varchar(20) NOT NULL,  
  `estoque` int(10) UNSIGNED NOT NULL,  
  `minimo` int(10) UNSIGNED NOT NULL,  
  `validade` date NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
```

```
-- -----
```

```
--  
-- Table structure for table `fornecedor`  
--
```

```
DROP TABLE IF EXISTS `fornecedor`;  
CREATE TABLE `fornecedor` (  
  `idfornecedor` int(10) UNSIGNED NOT NULL,  
  `nome` varchar(100) NOT NULL,  
  `cnpj` varchar(14) NOT NULL,  
  `telefone` varchar(11) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
```

```
-- -----
```

```
--  
-- Table structure for table `funcionarios`  
--
```

```
DROP TABLE IF EXISTS `funcionarios`;  
CREATE TABLE `funcionarios` (  
  `idfuncionario` int(10) UNSIGNED NOT NULL,  
  `nome_funcionario` varchar(25) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
```

--
-- Table structure for table `funcionarios_retira`
--

```
DROP TABLE IF EXISTS `funcionarios_retira`;  
CREATE TABLE `funcionarios_retira` (  
  `id` int(10) UNSIGNED NOT NULL,  
  `funcionarios_idfuncionario` int(10) UNSIGNED NOT NULL,  
  `epis_id` int(10) UNSIGNED NOT NULL,  
  `almoxarife_id` int(10) UNSIGNED NOT NULL,  
  `data_retirada` date NOT NULL,  
  `quantidade` int(10) UNSIGNED NOT NULL,  
  `devolvido` tinyint(1) DEFAULT 0,  
  `data_devolucao` date NOT NULL,  
  `comentario_devolucao` text DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
```

```
--  
-- Triggers `funcionarios_retira`  
--  
DROP TRIGGER IF EXISTS `TGR_devolve_epi`;  
DELIMITER $$  
CREATE TRIGGER `TGR_devolve_epi` AFTER UPDATE ON `funcionarios_retira`  
FOR EACH ROW begin
```

```
CALL atualiza_estoque(OLD.epis_id, old.quantidade, "D");
```

```
END  
$$  
DELIMITER ;  
DROP TRIGGER IF EXISTS `TRG_retirada_epi`;  
DELIMITER $$  
CREATE TRIGGER `TRG_retirada_epi` AFTER INSERT ON `funcionarios_retira`  
FOR EACH ROW BEGIN
```

```
CALL atualiza_estoque(NEW.epis_id, NEW.quantidade, "R");
```

```
END  
$$
```

DELIMITER ;

--

-- Indexes for dumped tables

--

--

-- Indexes for table `almoxarife`

--

ALTER TABLE `almoxarife`
ADD PRIMARY KEY (`id`);

--

-- Indexes for table `aviso`

--

ALTER TABLE `aviso`
ADD PRIMARY KEY (`idaviso`),
ADD KEY `alertas_FKIndex1` (`almoxarife_id`);

--

-- Indexes for table `compras`

--

ALTER TABLE `compras`
ADD PRIMARY KEY (`idcompra`),
ADD KEY `compras_FKIndex1` (`fornecedor_idfornecedor`),
ADD KEY `compras_FKIndex2` (`epis_id`);

--

-- Indexes for table `epis`

--

ALTER TABLE `epis`
ADD PRIMARY KEY (`id`);

--

-- Indexes for table `fornecedor`

--

ALTER TABLE `fornecedor`
ADD PRIMARY KEY (`idfornecedor`);

--

-- Indexes for table `funcionarios`


```
--  
ALTER TABLE `funcionarios`  
  ADD PRIMARY KEY (`idfuncionario`);  
  
--  
-- Indexes for table `funcionarios_retira`  
--  
ALTER TABLE `funcionarios_retira`  
  ADD PRIMARY KEY (`id`),  
  ADD KEY `funcionarios_retira_EPIs_FKIndex3` (`almoxarife_id`),  
  ADD KEY `funcionarios_retira_FKIndex2` (`epis_id`),  
  ADD KEY `funcionarios_retira_FKIndex3` (`funcionarios_idfuncionario`);  
  
--  
-- AUTO_INCREMENT for dumped tables  
--  
  
--  
-- AUTO_INCREMENT for table `almoxarife`  
--  
ALTER TABLE `almoxarife`  
  MODIFY `id` int(10) UNSIGNED NOT NULL AUTO_INCREMENT,  
  AUTO_INCREMENT=2;  
  
--  
-- AUTO_INCREMENT for table `aviso`  
--  
ALTER TABLE `aviso`  
  MODIFY `idaviso` int(10) UNSIGNED NOT NULL AUTO_INCREMENT;  
  
--  
-- AUTO_INCREMENT for table `compras`  
--  
ALTER TABLE `compras`  
  MODIFY `idcompra` int(10) UNSIGNED NOT NULL AUTO_INCREMENT;  
  
--  
-- AUTO_INCREMENT for table `epis`  
--  
ALTER TABLE `epis`  
  MODIFY `id` int(10) UNSIGNED NOT NULL AUTO_INCREMENT;
```

```
--
-- AUTO_INCREMENT for table `fornecedor`
--
ALTER TABLE `fornecedor`
  MODIFY `idfornecedor` int(10) UNSIGNED NOT NULL AUTO_INCREMENT;

--
-- AUTO_INCREMENT for table `funcionarios`
--
ALTER TABLE `funcionarios`
  MODIFY `idfuncionario` int(10) UNSIGNED NOT NULL AUTO_INCREMENT;

--
-- AUTO_INCREMENT for table `funcionarios_retira`
--
ALTER TABLE `funcionarios_retira`
  MODIFY `id` int(10) UNSIGNED NOT NULL AUTO_INCREMENT;

--
-- Constraints for dumped tables
--

--
-- Constraints for table `aviso`
--
ALTER TABLE `aviso`
  ADD CONSTRAINT `aviso_ibfk_1` FOREIGN KEY (`almoxarife_id`) REFERENCES
`almoxarife` (`id`) ON DELETE CASCADE ON UPDATE CASCADE;

--
-- Constraints for table `compras`
--
ALTER TABLE `compras`
  ADD CONSTRAINT `compras_ibfk_1` FOREIGN KEY (`fornecedor_idfornecedor`)
REFERENCES `fornecedor` (`idfornecedor`) ON DELETE CASCADE ON UPDATE
CASCADE,
  ADD CONSTRAINT `compras_ibfk_2` FOREIGN KEY (`epis_id`) REFERENCES
`epis` (`id`) ON DELETE CASCADE ON UPDATE CASCADE;

--
```

```
-- Constraints for table `funcionarios_retira`  
--  
ALTER TABLE `funcionarios_retira`  
  ADD CONSTRAINT `funcionarios_retira_ibfk_1` FOREIGN KEY (`almoxarife_id`)  
  REFERENCES `almoxarife` (`id`) ON DELETE CASCADE ON UPDATE CASCADE,  
  ADD CONSTRAINT `funcionarios_retira_ibfk_2` FOREIGN KEY (`epis_id`)  
  REFERENCES `epis` (`id`) ON DELETE CASCADE ON UPDATE CASCADE,  
  ADD CONSTRAINT `funcionarios_retira_ibfk_3` FOREIGN KEY  
  (`funcionarios_idfuncionario`) REFERENCES `funcionarios` (`idfuncionario`) ON  
  DELETE CASCADE ON UPDATE CASCADE;  
COMMIT;  
  
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;  
/*!40101 SET  
CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;  
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```

3.7 Recursos e ferramentas (1º e 2º)

[VISUAL STUDIO CODE](#) foi usado para editar o código.

[XAMPP](#) foi usado como uma interface acessar aos servidores [php](#) e [mysql](#)

[BOOTSTRAP](#) é um framework web que facilita o desenvolvimento de componentes de interface.

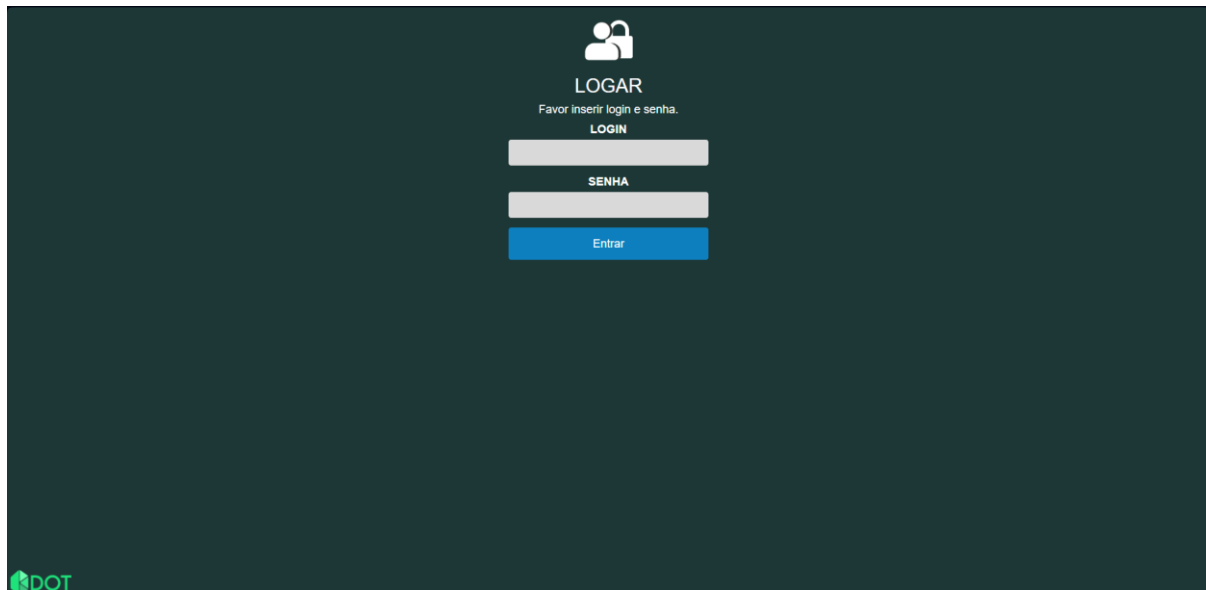
[GITHUB](#) é uma plataforma que auxiliou no processo de versionamento.

[PLANTUML](#) auxiliou no desenvolvimento de diagramas.

3.8 Tela

Interface da página para realizar o Login

Fatec Araras "Antônio Brambilla"



The login interface features a dark teal background. At the top center is a white icon of two people. Below it, the word "LOGAR" is displayed in white. A smaller line of text reads "Favor inserir login e senha." followed by the word "LOGIN" in all caps. There are two white input fields for login and password, each with a label "LOGIN" and "SENHA" respectively. A blue button labeled "Entrar" is positioned below the password field. In the bottom left corner, there is a green "DOT" logo.

Interface da página de "Menu" do Usuário "adm" (um usuário do tipo supervisor)



The user menu interface has a dark teal background. In the top left corner, it says "Usuário: adm". In the top right corner, there is a red button labeled "Sair da Conta". At the top center is a white house icon. Below it, the text "SEJA BEM-VINDO" is displayed, followed by "O que deseja fazer hoje?". A vertical list of buttons is centered on the page: "Histórico de Devoluções", "Histórico de Saídas", "Ver Estoque", "Cadastrar Almoxarife", "Cadastrar Fornecedor", "Cadastrar Funcionário", "Registrar Compra", and "Checar Alertas". The "Checar Alertas" button is orange, while the others are green. A green "DOT" logo is in the bottom left corner, and a bell icon with a notification count is in the bottom right corner.

Interface da página "Histórico de Devoluções"

Fatec Araras "Antônio Brambilla"




Interface da página "Histórico de Saídas"



Interface da página "Ver Estoque"


Fatec Araras "Antônio Brambilla"


Lista de estoque
Procurar E.P.I


Pesquisar

Voltar

ID	Nome	C.A	Unidade	Estoque	Mínimo	Validade
----	------	-----	---------	---------	--------	----------




Interface da página "Cadastrar Almoxarife"


CADASTRAR
Favor inserir usuário e senha.
USUARIO

SENHA

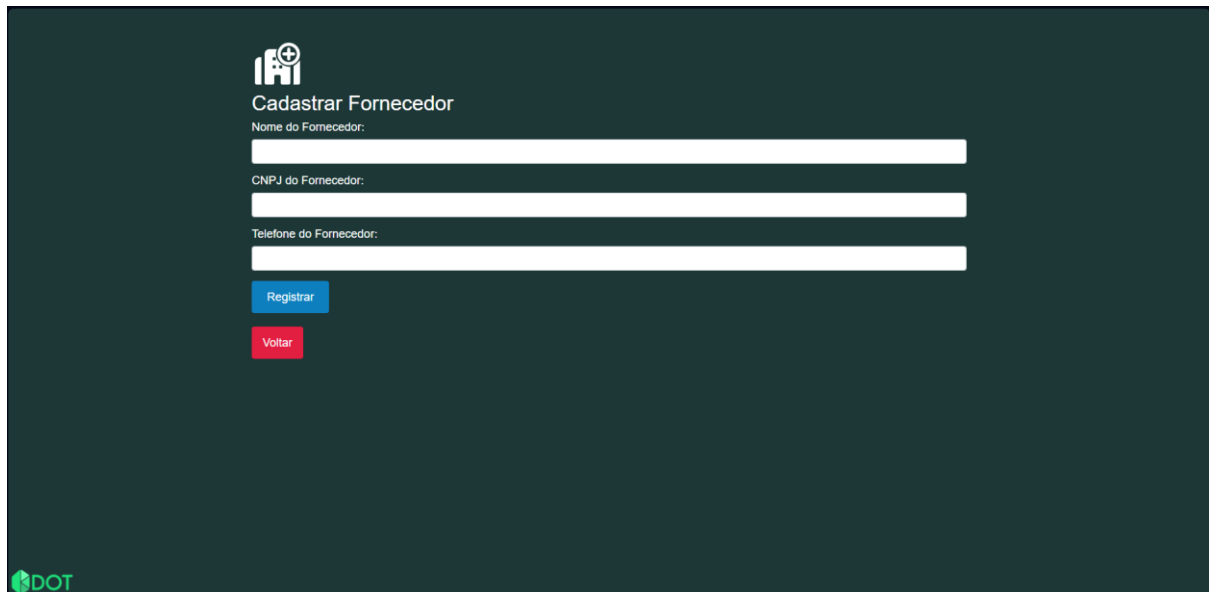
Enviar


Voltar



Interface da página "Cadastrar Fornecedor"

Fatec Araras "Antônio Brambilla"





Cadastrar Fornecedor


Nome do Fornecedor:

CNPJ do Fornecedor:

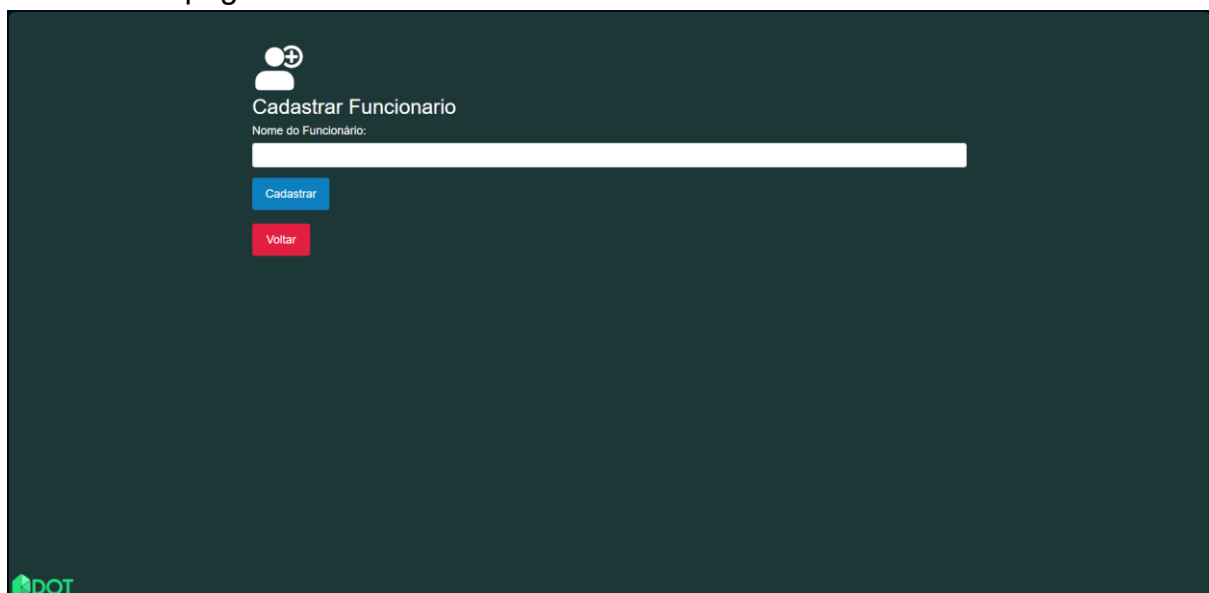
Telefone do Fornecedor:


Registrar

Voltar



Interface da página "Cadastrar Funcionário"






Cadastrar Funcionario

Nome do Funcionário:

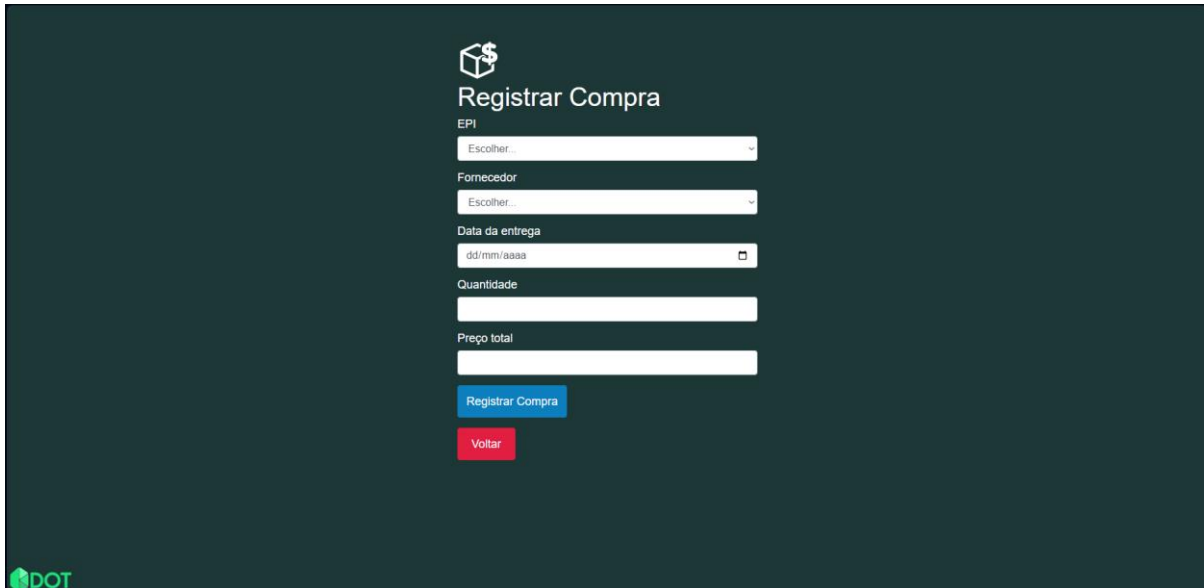
Cadastrar


Voltar



Interface da página "Registrar Compra"

Fatec Araras “Antônio Brambilla”




Registrar Compra

EPI

Fornecedor


Data da entrega

Quantidade

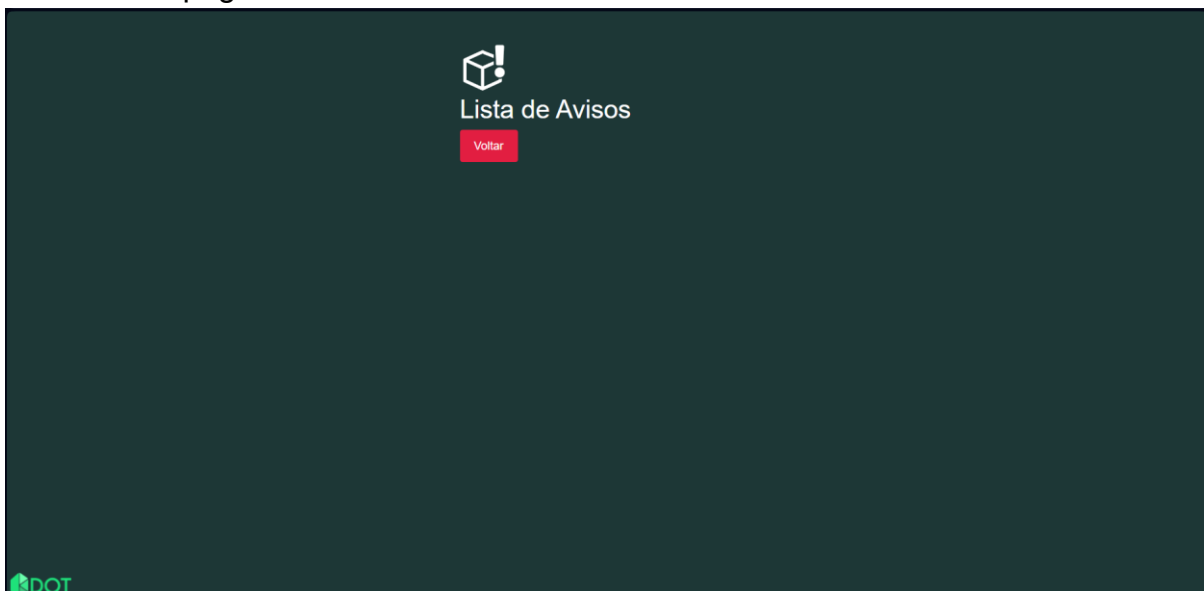
Preço total


[Registrar Compra](#)

[Voltar](#)




Interface da página “Checar Alertas”




Lista de Avisos

[Voltar](#)



Interface da página “Menu” do usuário “almoxarife” (um usuário do tipo normal)

Fatec Araras "Antônio Brambilla"

Usuário: almozarife Sair da Conta




SEJA BEM-VINDO
O que deseja fazer hoje ?

- Histórico de Devoluções
- Histórico de Saídas
- Ver Estoque
- Registrar Devolução
- Registrar Saída
- Adicionar EPI ao Estoque
- Remover EPI
- Enviar Alerta



Interface da página "Registrar Devolução"




Registrar Devolução de EPI

ID da Retirada:

Comentário (opcional):

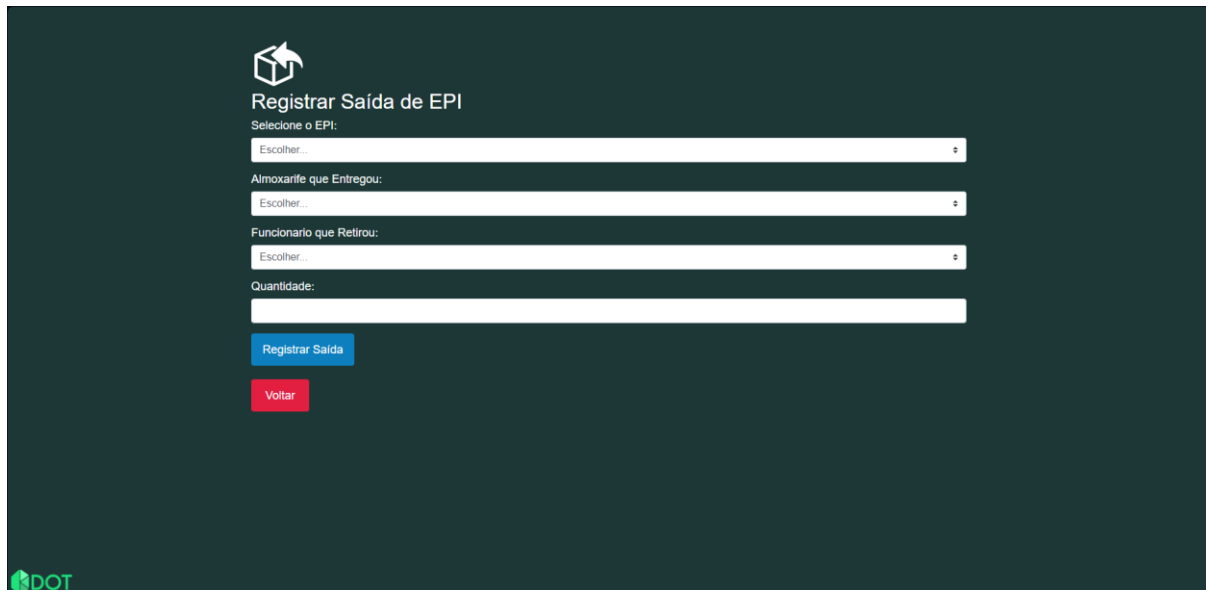
Registrar Devolução

Voltar



Interface da página "Registrar Saída"

Fatec Araras "Antônio Brambilla"

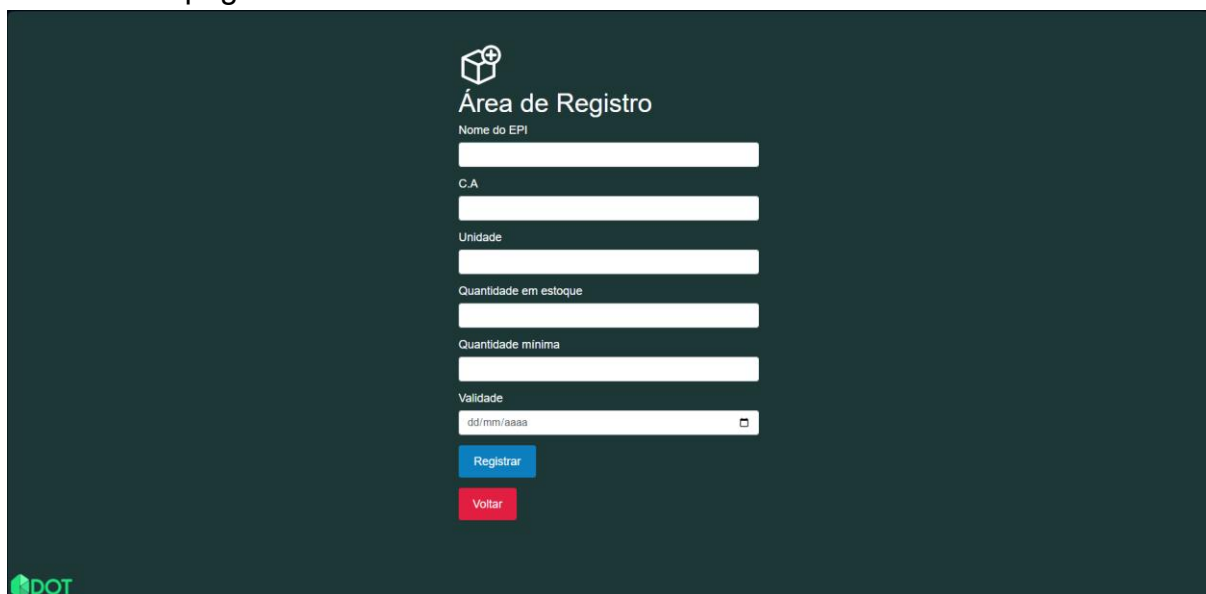


The screenshot shows a web form titled "Registrar Saída de EPI" (Register Exit of PPE) on a dark green background. At the top left is a small icon of a box with an arrow. The form contains the following fields and buttons:

- Seleção o EPI:** A dropdown menu with the placeholder text "Escolher..." and a small downward arrow.
- Almoxarife que Entregou:** A dropdown menu with the placeholder text "Escolher..." and a small downward arrow.
- Funcionário que Retirou:** A dropdown menu with the placeholder text "Escolher..." and a small downward arrow.
- Quantidade:** A text input field.
- Registrar Saída:** A blue button.
- Voltar:** A red button.

In the bottom left corner, there is a small logo for "DOT".

Interface da página "Adicionar EPI"



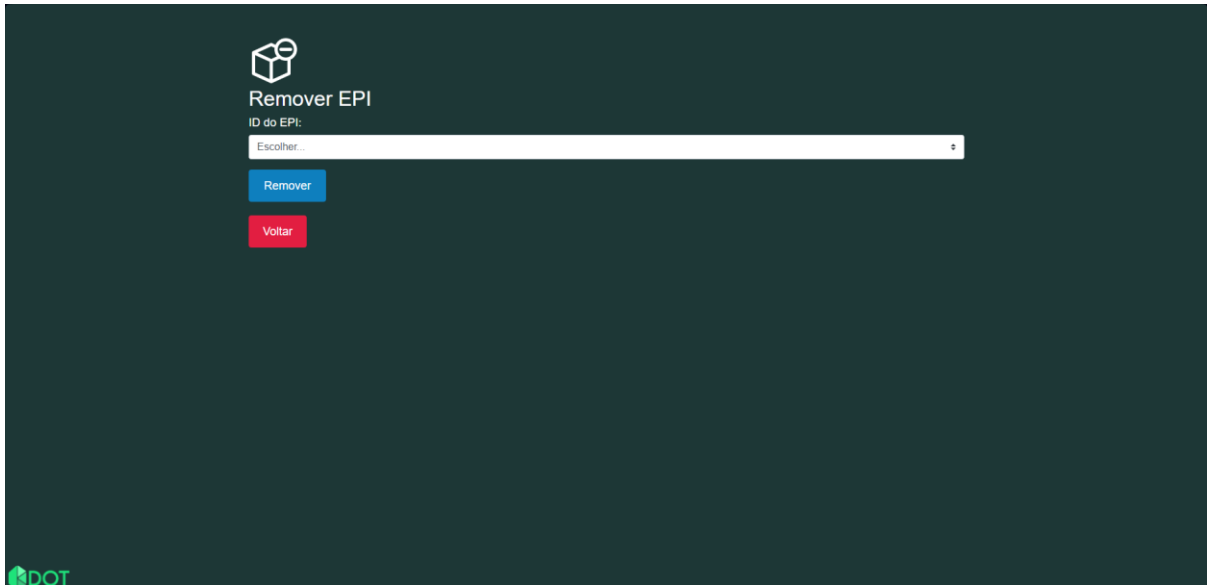
The screenshot shows a web form titled "Área de Registro" (Registration Area) on a dark green background. At the top left is a small icon of a box with a plus sign. The form contains the following fields and buttons:

- Nome do EPI:** A text input field.
- C.A:** A text input field.
- Unidade:** A text input field.
- Quantidade em estoque:** A text input field.
- Quantidade mínima:** A text input field.
- Validade:** A date input field with the placeholder text "dd/mm/aaaa" and a small calendar icon.
- Registrar:** A blue button.
- Voltar:** A red button.

In the bottom left corner, there is a small logo for "DOT".

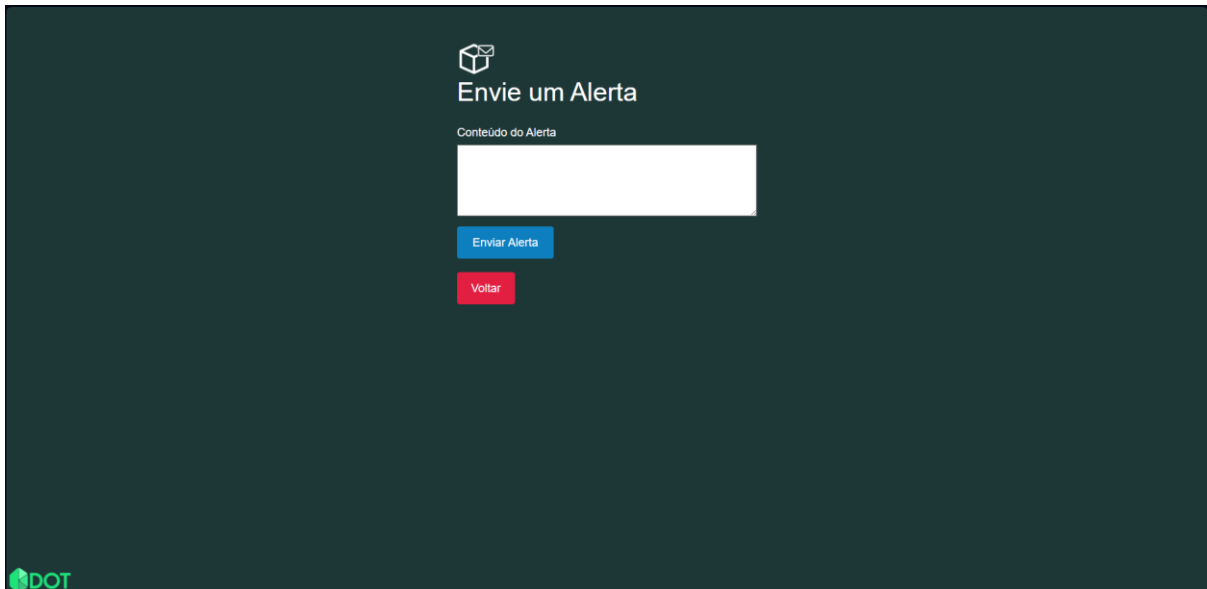
Interface da página "Remover EPI"

Fatec Araras "Antônio Brambilla"



The screenshot shows a web interface with a dark teal background. At the top left is a white icon of a cube with a circle inside. Below it, the text "Remover EPI" is displayed. Underneath, "ID do EPI:" is followed by a white input field containing the placeholder text "Escolher...". Below the input field are two buttons: a blue "Remover" button and a red "Voltar" button. In the bottom left corner, there is a small green logo with the text "DOT".

Interface da página "Enviar Alerta"



The screenshot shows a web interface with a dark teal background. At the top center is a white icon of a cube with a circle inside. Below it, the text "Envie um Alerta" is displayed. Underneath, "Conteúdo do Alerta" is followed by a white input field. Below the input field are two buttons: a blue "Enviar Alerta" button and a red "Voltar" button. In the bottom left corner, there is a small green logo with the text "DOT".

4. Testes e Qualidade

4.1 Estratégia de Testes: Descrever a estratégia de testes adotada

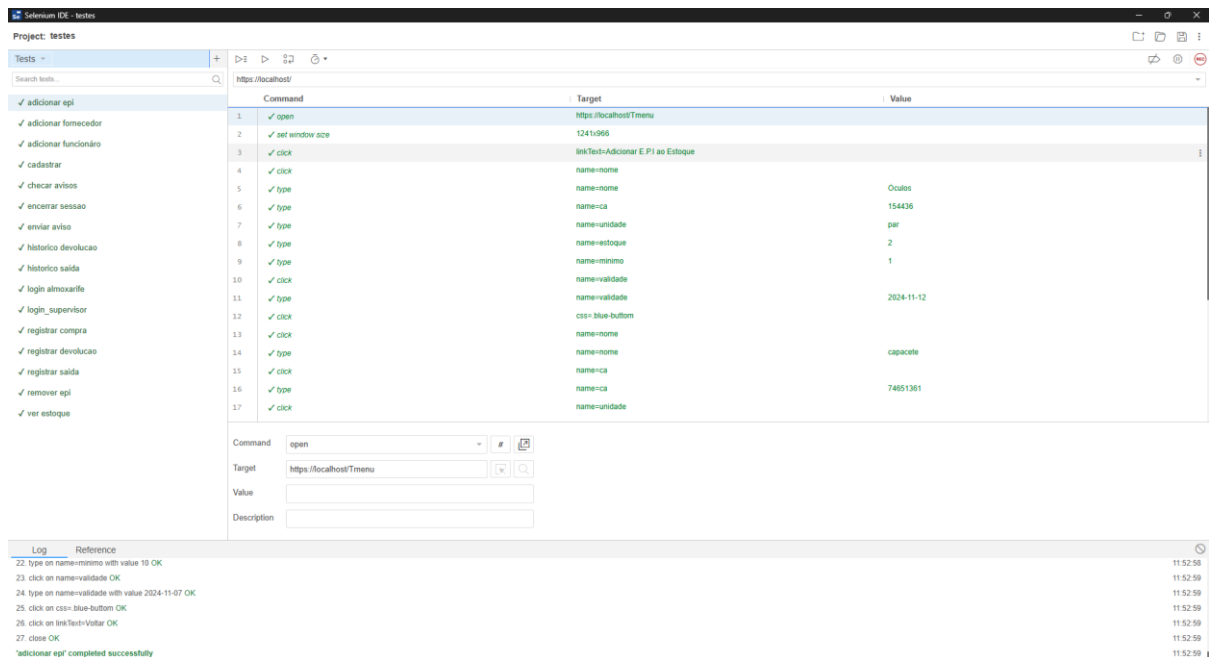
Teste de unidade: Após o término das etapas de desenvolvimento as classes e funções criadas foram extensamente testadas, com o propósito de garantir o comportamento adequado das funcionalidades.

Teste de Validação: A ferramenta Selenium foi usado para automatizar os testes, com o objetivo de verificar as entradas e saídas conhecidas pelo usuário estavam de acordo com o esperado. O Selenium utilizado foi a versão de extensão de browser.

4.2 Resultados dos Testes: Apresentar os resultados dos testes realizados

Resultado dos testes unitários: Todas as funções presentes demonstraram funcionamento dentro do planejado, performando suas chamadas e comunicações internas de forma correta.

Resultado dos Testes de Validação usando Selenium



Test Step	Command	Target	Value
1	open	https://localhost/	
2	set window size	1241x966	
3	click	linkText=Adicionar EPI ao Estoque	
4	click	name=name	
5	type	name=name	Oculos
6	type	name=ca	154436
7	type	name=unidade	par
8	type	name=estoque	2
9	type	name= minimo	1
10	click	name=validade	
11	type	name=validade	2024-11-12
12	click	css=blue-button	
13	click	name=name	
14	type	name=name	capacete
15	click	name=ca	
16	type	name=ca	74051361
17	click	name=unidade	

Log Reference

- 22. type on name= minimo with value 10 OK 11:52:59
- 23. click on name=validade OK 11:52:59
- 24. type on name=validade with value 2024-11-07 OK 11:52:59
- 25. click on css=blue-button OK 11:52:59
- 26. click on linkText=Visitante OK 11:52:59
- 27. close OK 11:52:59
- *adicionar epi completed successfully 11:52:59

4.3 Garantia da Qualidade: Descrever as práticas adotadas para garantia da qualidade

Funcionalidade: Foram realizadas reuniões com o cliente para entender os requisitos e principais objetivos da aplicação do software no ambiente de trabalho da empresa. Após a realização das funções, eram feitos diversos testes para validar sua funcionalidade.

Manutenibilidade: Utilizamos a convenção de nomes para facilitar a leitura e entendimento do código, juntamente com comentários explicando as funções.

Utilizamos o padrão de projeto MVC (Model-View-Controller) que facilita a manutenção e a reutilização do código.

Usabilidade: Foi desenvolvida uma interface simples com cores sóbrias, para que seja de fácil entendimento para o usuário, com o objetivo na agilidade, sem muitas burocracias.

4.4 Requisitos mínimos de hardware e software para o sistema

Requisitos mínimos de hardware:

- Processador de 2 GHz ou superior
- Pelo menos 2 GB de memória RAM, mas é recomendado 4 GB para melhor desempenho
- Cerca de 1 GB de espaço livre no disco
- Acesso às funções do administrador para instalar e configurar os componentes

Requisitos mínimos de software:

- Windows Vista e posteriores

4.5 Contrato para desenvolvimento de software

CONTRATO DE VENDA DE SOFTWARE

Pelo presente instrumento particular, as partes:

VENDEDOR: DOT Soluções Tecnológicas Ltda., com sede em [endereço completo], inscrita no CNPJ sob nº [número do CNPJ], doravante denominada "**Vendedor**";

COMPRADOR: [Nome da Empresa Compradora], com sede em [endereço completo], inscrita no CNPJ sob nº [número do CNPJ], doravante denominada "**Comprador**";

têm entre si, justas e contratadas, as cláusulas e condições seguintes:

CLÁUSULA 1 - OBJETO

1.1 O **Vendedor** compromete-se a vender e o **Comprador** compromete-se a adquirir o software denominado **DOT Gestão de Almoxarife**, doravante denominado "**Software**", que possui as seguintes características:

- **Nome do Software:** DOT Gestão de Almoxarife
- **Versão:** [Versão do Software]
- **Descrição:** O **Software DOT Gestão de Almoxarife** tem como objetivo otimizar a gestão de estoques e almoxarifado, proporcionando controle eficiente de entradas, saídas, movimentação de materiais e inventários.
- **Licença de uso:** Licença anual, **não exclusiva e intransferível**.

1.2 O **Software** será entregue ao **Comprador** de acordo com as especificações acordadas, conforme a proposta comercial anexa, que faz parte integrante deste contrato.

CLÁUSULA 2 - VALOR E FORMA DE PAGAMENTO

2.1 O valor total para a aquisição do **Software** será de **R\$ 3.000,00** (três mil reais), a ser pago em **5 (cinco) parcelas** de **R\$ 600,00** (seiscentos reais) cada.

2.2 As parcelas serão pagas nas seguintes condições:

- **1ª Parcela:** R\$ 600,00 (seiscentos reais) na assinatura deste contrato.
- **2ª a 5ª Parcelas:** R\$ 600,00 (seiscentos reais) cada, com vencimento a cada 30 (trinta) dias, a partir da assinatura deste contrato.

2.3 O pagamento será realizado por meio de [boleto bancário, transferência bancária, ou outra forma de pagamento acordada pelas partes].

2.4 Em caso de atraso no pagamento, será acrescido ao valor da parcela devida, juros de [informar %] ao mês e multa de [informar %] sobre o valor da parcela em atraso.

CLÁUSULA 3 - ENTREGA E INSTALAÇÃO

3.1 O **Vendedor** compromete-se a entregar o **Software** ao **Comprador** até a data de [data de entrega], por meio de [método de entrega, como link para download ou mídia física].

3.2 O **Vendedor** prestará serviços de instalação e configuração do **Software** no ambiente do **Comprador**, caso necessário, mediante custo adicional de R\$ [valor de instalação/configuração], conforme acordado entre as partes.

CLÁUSULA 4 - LICENÇA DE USO

4.1 O **Vendedor** concede ao **Comprador** uma licença de uso do **Software DOT Gestão de Almoxarife, não exclusiva, intransferível**, válida por 12 (doze) meses, com direito a renovação conforme o interesse do **Comprador**.

4.2 O **Comprador** não poderá sublicenciar, distribuir, modificar ou revender o **Software** sem a prévia autorização por escrito do **Vendedor**.

CLÁUSULA 5 - SUPORTE E ATUALIZAÇÕES

5.1 O **Vendedor** fornecerá ao **Comprador** suporte técnico referente ao **Software** pelo período de 12 (doze) meses, contado a partir da data de entrega, de acordo com a política de suporte descrita na proposta comercial.

5.2 O **Vendedor** também fornecerá atualizações do **Software**, incluindo melhorias e correções de erros, conforme sua política interna.

CLÁUSULA 6 - OBRIGAÇÕES DAS PARTES

6.1 Obrigações do Vendedor:

- Entregar o **Software** conforme as especificações acordadas.
- Prestar suporte técnico e manutenção conforme descrito na Cláusula 5.
- Fornecer as atualizações do **Software**, conforme a política de atualizações.

6.2 Obrigações do Comprador:

- Efetuar o pagamento do valor acordado, nas condições e prazos estabelecidos.
- Utilizar o **Software** conforme os termos da licença de uso.
- Notificar o **Vendedor** sobre qualquer falha ou erro no **Software** para que as correções sejam feitas.

CLÁUSULA 7 - GARANTIA

7.1 O **Vendedor** garante que o **Software** estará livre de defeitos de fabricação por um período de 12 (doze) meses, a contar da data de entrega.

7.2 Caso o **Software** apresente falhas ou erros durante o período de garantia, o **Vendedor** compromete-se a corrigir tais falhas sem custos adicionais para o **Comprador**, desde que tais falhas não sejam causadas por uso inadequado ou modificações realizadas pelo **Comprador**.

CLÁUSULA 8 - RESCISÃO

8.1 Este contrato poderá ser rescindido por qualquer das partes, sem ônus, nas seguintes hipóteses:

- Descumprimento de qualquer cláusula contratual por uma das partes, após notificação formal e prazo de 15 (quinze) dias para correção.
- Não pagamento das parcelas por parte do **Comprador**, conforme estipulado na Cláusula 2.

8.2 Em caso de rescisão, o **Vendedor** poderá exigir o pagamento das parcelas vencidas e não pagas até a data da rescisão.

CLÁUSULA 9 - CONFIDENCIALIDADE

9.1 Ambas as partes se comprometem a manter em sigilo todas as informações confidenciais compartilhadas durante a vigência deste contrato, incluindo, mas não se limitando a, detalhes técnicos do **Software**, condições comerciais e dados financeiros.

CLÁUSULA 10 - DISPOSIÇÕES GERAIS

10.1 O **Comprador** não poderá ceder ou transferir este contrato, ou seus direitos e obrigações, sem a prévia autorização por escrito do **Vendedor**.

Fatec Araras "Antônio Brambilla"

10.2 Caso alguma cláusula deste contrato seja considerada inválida ou inaplicável, as demais cláusulas permanecerão em pleno vigor.

10.3 Este contrato é regido pelas leis da República Federativa do Brasil.

CLÁUSULA 11 - FORO

11.1 Fica eleito o foro da comarca de [cidade], estado de [estado], para dirimir quaisquer controvérsias oriundas deste contrato, com renúncia a qualquer outro, por mais privilegiado que seja.

Por estarem assim justas e contratadas, as partes assinam este instrumento em duas vias de igual teor e forma, na presença das testemunhas abaixo.

[Local], [Data]

Vendedor

Nome: [Nome do Representante do Vendedor]

Cargo: [Cargo do Representante]

Assinatura: _____

Comprador

Nome: [Nome do Representante do Comprador]

Cargo: [Cargo do Representante]

Assinatura: _____

Testemunhas:

1. Nome: _____

CPF: _____

Assinatura: _____

2. Nome: _____

CPF: _____

Assinatura: _____

3. Considerações Finais

Conclusão: O projeto está de acordo com o planejado, o que é de grande satisfação para toda a equipe de desenvolvimento. Aprendemos juntos durante o processo que é necessário paciência e persistência para obter êxito e assim entregamos um software de gestão de almoxarife totalmente funcional. Valorizamos o aprendizado de saber como nos organizarmos de forma eficiente para que pudéssemos cumprir o prazo sem fadiga.

Contribuições Individuais:

Davi Bonelli – Back-End.

Davi Samuel – Front-End.

João Pedro – Coordenou o projeto como PO e Desenvolvedor Back-End

Izabela Kerollayne - Documentação.

6. Referências

- <https://getbootstrap.com.br/>
- <https://www.w3schools.com/>
- <https://www.php.net/>

Anexo I - Diário de bordo

Físico: Feito pelo grupo.

Digital: https://github.com/jp8002/PI_FATEC_2024_2-Semestre/blob/main/diario.md

Anexo II – Cronograma efetivo

O cronograma foi executado conforme o planejado, conseguimos cumprir com os prazos que estipulamos.

Anexo III – Evidências

Nada a ser comentado.