



UFOP

Universidade Federal
de Ouro Preto

**Universidade Federal de Ouro Preto
Instituto de Ciências Exatas e Aplicadas
Departamento de Computação e Sistemas**

**Armazém Web: Aplicação web com
conexão ao ERP Protheus para
Requisições ao Armazém**

Emily Ferreira de Brito

**TRABALHO DE
CONCLUSÃO DE CURSO**

ORIENTAÇÃO:

Filipe Nunes Ribeiro

COORIENTAÇÃO:

Vicente J. P. Amorim

**Novembro, 2022
João Monlevade–MG**

Emily Ferreira de Brito

**Armazém Web: Aplicação web com conexão ao
ERP Protheus para Requisições ao Armazém**

Orientador: Filipe Nunes Ribeiro

Coorientador: Vicente J. P. Amorim

Monografia apresentada ao curso de Engenharia de Computação do Instituto de Ciências Exatas e Aplicadas, da Universidade Federal de Ouro Preto, como requisito parcial para aprovação na Disciplina “Trabalho de Conclusão de Curso II”.

Universidade Federal de Ouro Preto

João Monlevade

Novembro de 2022

SISBIN - SISTEMA DE BIBLIOTECAS E INFORMAÇÃO

B862a Brito, Emily Ferreira de.
Armazém web [manuscrito]: aplicação web com conexão ao ERP
Protheus para requisições ao armazém. / Emily Ferreira de Brito. - 2022.
78 f.: il.: color., gráf..

Orientador: Prof. Dr. Filipe Nunes Ribeiro.
Monografia (Bacharelado). Universidade Federal de Ouro Preto.
Instituto de Ciências Exatas e Aplicadas. Graduação em Engenharia de
Computação .

1. Aplicações Web. 2. Controle de estoque. 3. Integração de aplicações
corporativas (Sistemas de computação). 4. Sistemas de informação
gerencial. I. Ribeiro, Filipe Nunes. II. Universidade Federal de Ouro Preto.
III. Título.

CDU 658:004

Bibliotecário(a) Responsável: Flávia Reis - CRB6-2431



FOLHA DE APROVAÇÃO

Emily Ferreira de Brito

Armazém web: aplicação web com conexão ao ERP Protheus para requisições ao armazém

Monografia apresentada ao Curso de Engenharia de Computação da Universidade Federal de Ouro Preto como requisito parcial para obtenção do título de Bacharel em Engenharia de Computação

Aprovada em 03 de novembro de 2022

Membros da banca

Doutor - Filipe Nunes Ribeiro - Orientador (Universidade Federal de Ouro Preto)
Doutora - Helen de Cássia Sousa da Costa Lima - (Universidade Federal de Ouro Preto)
Doutor - Fernando Bernardes de Oliveira - (Universidade Federal de Ouro Preto)

Filipe Nunes Ribeiro, orientador do trabalho, aprovou a versão final e autorizou seu depósito na Biblioteca Digital de Trabalhos de Conclusão de Curso da UFOP em 07/12/2022



Documento assinado eletronicamente por **Filipe Nunes Ribeiro, PROFESSOR DE MAGISTERIO SUPERIOR**, em 07/12/2022, às 11:21, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site http://sei.ufop.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **0439692** e o código CRC **A7BCE539**.

Este trabalho é dedicado especialmente a minha mãe Valéria que não mediu esforços para que esta conquista acontecesse, sempre me apoiou e esteve ao meu lado em todos os momentos. A minha avó Maria que possibilitou que essa trajetória fosse possível e ao meu avô Paulo pela oração e carinho.

Agradecimentos

Agradeço primeiramente a Deus por permitir que essa caminhada fosse traçada, pelo acalento e provações.

A minha mãe Valéria por estar sempre ao meu lado e apoiar as minhas decisões, por se fazer presente mesmo estando longe e por acreditar mais em mim que eu mesma.

Agradeço aos colegas de turma e de moradia, que se tornaram alicerce e companhia durante esta jornada.

Agradeço a minha equipe de trabalho, que apoiaram e acreditaram neste trabalho, além de proporcionarem todo o auxílio necessário.

Agradeço ainda ao professor Vicente, que acreditou e incentivou a ideia deste trabalho, e ao professor Filipe que se propôs a me auxiliar no que foi necessário.

Por fim, agradeço a todos que contribuíram para meu crescimento pessoal, profissional e acadêmico que aqui não foram pontualmente citados.

“A tecnologia move o mundo.”

— Steve Jobs, inventor e fundador da Apple.

Resumo

Para suprir a necessidade de integrar e gerenciar seus processos de forma que consiga visualizar a empresa como um todo, as empresas aderiram aos sistemas gerenciais ERP. Entre esses ERPs, está o Protheus que possui um sistema robusto e de grande capacidade, atende diversos segmentos e alcança um público extenso, porém possui algumas desvantagens, como limitação de acesso por gestão de licenças e interface complexa e pouco amigável. Visto que a gestão e controle de estoque é de grande importância e a requisição ao armazém gera alta demanda para os funcionários do setor de estoque, o trabalho propõe a adesão da solução padrão do Protheus para requisições ao armazém de forma econômica através de uma aplicação web que consome apenas a uma licença de serviço REST, mas permite acesso ilimitado de usuários, e se conecta ao Protheus através de API. A aplicação utiliza o banco de dados do Protheus e mantém o conceito de integração de ERP, além de oferecer uma interface amigável e objetiva para atender o público diversificado de funcionários de uma empresa.

Palavras-chaves: ERP. protheus. estoque. web. API.

Abstract

To meet the need to integrate and manage their processes in a way that can visualize the company as a whole, companies have joined ERP management systems. Among these ERPs, there is Protheus, which has a robust and high-capacity system, serving different segments and reaching an extensive audience. However, it has some disadvantages, such as limited access due to license management and a complex and unfriendly interface. Since the management and control of stock is of great importance and the requisition to the warehouse generates high demand for the warehouse employees, the work proposes the adhesion of the standard Protheus solution for requisitions to the warehouse in an economical way through a web application that consumes only a REST service license, but it allows unlimited user access, and connects to Protheus through an API. The application uses the Protheus database and maintains the concept of ERP integration, in addition to offering a friendly and objective interface to serve the diverse public of employees of a company.

Key-words: ERP. protheus. inventory. web. API.

Lista de ilustrações

Figura 1 – Fluxo de Requisição ao Armazém Padrão do Protheus.	19
Figura 2 – Fluxo do processo de requisição ao armazém do Estudo de Caso.	27
Figura 3 – Caso de uso do protótipo.	28
Figura 4 – Fluxo do Caso de Uso do Protótipo.	29
Figura 5 – Protótipo da página inicial.	29
Figura 6 – Protótipo da página principal.	30
Figura 7 – Protótipo da página de produtos.	31
Figura 8 – Protótipo da página vazia do carrinho de requisições.	32
Figura 9 – Protótipo da página de preenchimento do carrinho de requisições.	33
Figura 10 – Diagrama de Casos de Uso.	35
Figura 11 – Diagrama de Atividade referente a Edição de uma Requisição.	36
Figura 12 – Diagrama de Sequência de Incluir Nova Requisição.	37
Figura 13 – Modelo Relacional do Banco de Dados.	38
Figura 14 – Estruturação das APIS.	40
Figura 15 – Versionamento no <i>GitHub</i>	41
Figura 16 – Exemplos das consultas no Postman.	43
Figura 17 – Fluxo geral do processo implementado.	44
Figura 18 – Tela de <i>login</i> da aplicação.	45
Figura 19 – Tela de cadastro da aplicação.	45
Figura 20 – Tela de alteração de senha da aplicação.	46
Figura 21 – Tela inicial da aplicação.	47
Figura 22 – Página de Requisições.	48
Figura 23 – Itens das requisições.	49
Figura 24 – Janela de Legenda dos status de requisições.	49
Figura 25 – Fluxo de status da requisição.	50
Figura 26 – Janela de filtros da Página de Requisições.	50
Figura 27 – Página de Requisições com filtro aplicado.	51
Figura 28 – Página de Requisições.	51
Figura 29 – Página de Produtos.	52
Figura 30 – Janela de filtros da Página de Produtos.	53
Figura 31 – Página do Carrinho de Requisições.	53
Figura 32 – Seletor de Centro de Custos.	53
Figura 33 – Página de edição.	54
Figura 34 – Página de edição apenas no modo visualização.	55
Figura 35 – Janela de confirmação para excluir requisição.	55
Figura 36 – Erro ao excluir requisição.	56

Figura 37 – Diagrama de Atividade de Cadastro de Usuário.	62
Figura 38 – Diagrama de Atividade de Alterar Senha de Usuário.	62
Figura 39 – Diagrama de Atividade de Incluir Nova Requisição.	62
Figura 40 – Diagrama de Atividade de Excluir Requisição.	63
Figura 41 – Diagrama de Sequência de Cadastrar no Usuário e Alterar Senha.	63
Figura 42 – Diagrama de Sequência de Login e Visualizar Requisições.	64
Figura 43 – Diagrama de Sequência de Edição.	64
Figura 44 – Diagrama de Sequência de Exclusão.	64
Figura 45 – Configuração para bloquear campo Solicitante.	65
Figura 46 – Criação do campo Requisiteante na tabela SCP	66
Figura 47 – Criação do campo Descrição do Requisiteante na tabela SCP	67
Figura 48 – Campos da tabela SZZ.	68
Figura 49 – Campo Filial da tabela SZZ.	68
Figura 50 – Campo User da tabela SZZ.	69
Figura 51 – Campo Matrícula da tabela SZZ.	70
Figura 52 – Campo Nome da tabela SZZ.	71
Figura 53 – Campo Senha da tabela SZZ.	72
Figura 54 – Campo Bloqueio da tabela SZZ.	73
Figura 55 – Campos Data Fim da tabela SZZ.	74
Figura 56 – Índice da tabela SZZ.	74
Figura 57 – Configuração do menu para incluir rotina de Usuários do Aramzém.	75
Figura 58 – Exemplo de bloqueio de usuário do armazém pela rotina criada no Protheus.	76
Figura 59 – Janela de confirmação para Gerar Pré-Requisito no Protheus.	76
Figura 60 – Janela de definição de parâmetros para a Geração de Pré-Requisição no Protheus.	77
Figura 61 – Rotinas no menu do Protheus.	78

Lista de abreviaturas e siglas

ERP *Enterprise Resource Planning*

API *Application Programming Interface*

JSON *Javascript Object Notation*

REST *Representational State Transfer*

UML *Unified Modeling Language*

CRUD *Create, Read, Update, Delete*

MVC *Model-View-Controller*

ISS Imposto Sobre Serviços

IPI Imposto sobre Produtos Industrializados

ICMS Imposto sobre Circulação de Mercadorias e Prestações de Serviços

ST Imposto Sobre Serviços

MRP I *Manufacturing Resource Planning I*

MRP II *Manufacturing Resource Planning II*

Sumário

1	INTRODUÇÃO	14
1.1	Objetivos	16
1.2	Metodologia	17
1.3	Organização do trabalho	17
2	FUNDAMENTAÇÃO TEÓRICA	18
2.1	ERP TOTVS <i>Backoffice</i> - Linha Protheus	18
2.1.1	Módulo Estoque e Custos	18
2.2	API REST	19
2.3	Trabalhos Relacionados	20
2.3.1	A importância do uso de <i>software</i> ERP como ferramenta para um eficiente controle de estoque	20
2.3.2	Planejamento e Controle de Estoque nas Organizações.	21
2.3.3	ERP - Módulo ESTOQUE/CUSTOS, PCP e os problemas burocráticos	22
2.3.4	Conclusão	23
3	DESENVOLVIMENTO	24
3.1	Ferramentas e tecnologias	24
3.1.1	Linguagens	24
3.1.2	<i>Laravel</i>	24
3.1.3	<i>Bootstrap</i>	25
3.1.4	<i>jQuery</i>	25
3.1.5	AdminLTE	25
3.1.6	SQL Server	25
3.1.7	<i>GitHub</i>	25
3.1.8	Mural	25
3.1.9	<i>Astah UML</i>	26
3.1.10	<i>Postman</i>	26
3.1.11	<i>VS Code</i>	26
3.2	Estudo de Caso	26
3.3	Modelagem	34
3.3.1	Requisitos	34
3.3.1.1	Requisitos Funcionais	34
3.3.1.2	Requisitos Não Funcionais	34
3.3.2	Casos de Uso	35
3.3.3	Diagrama de Atividade	36

3.3.4	Diagrama de Sequência	36
3.3.5	Banco de Dados	36
3.3.6	Arquitetura do Software	39
3.4	Implementação	39
3.4.1	Pré-Requisito	39
3.4.2	APIs	39
3.4.3	Interface Gráfica	41
3.4.4	Versionamento	41
3.5	Testes	42
3.5.1	Testes das APIs	42
3.5.2	Avaliação da Experiência do Usuário	42
4	RESULTADOS	44
4.1	<i>Login</i>	44
4.2	Página Inicial	46
4.3	Página de Requisições	47
4.4	Inclusão de uma Nova Requisição	51
4.5	Editar e excluir uma requisição	54
5	CONCLUSÃO	57
5.1	Trabalhos Futuros	58
	REFERÊNCIAS	59
	APÊNDICES	61
	APÊNDICE A – DIAGRAMAS COMPLEMENTARES	62
A.1	Diagrama de Atividade	62
A.2	Diagrama de Sequência	63
	APÊNDICE B – MODELO RELACIONAL	65
B.1	Adequações	65

1 Introdução

As empresas estão inseridas em um ambiente onde são constantemente afetadas por aspectos externos, como fatores econômicos, políticos, sociais e tecnológicos, fazendo com a que a competitividade esteja em constante crescimento. Isso leva as empresas a uma situação em que estão sempre em busca de mudanças e adaptações para diminuir custos e encontrar diferenciações para seus produtos e serviços.

Para se manter em uma posição de competitividade e evolução, as empresas se depararam com a necessidade de integrar e gerenciar seus processos de forma que o gestor da empresa consiga visualizar a empresa como um todo sob a perspectiva funcional e sob a perspectiva sistêmica, facilitando o acesso a informações e dados da empresa, de forma assertiva e pontual e trazendo benefícios para os processos internos.

Como forma de suprir essa demanda surgiram os sistemas gerenciais *Enterprise Resource Planning* (ERP), um sistema informatizado responsável por cuidar de todas as operações administrativas e operacionais da empresa. Um benefício deste software é a integração de diversas áreas, como orçamento, execução, controladoria, finanças, gerenciamento de projetos, controle de estoque e recursos humanos. (PAVIN; KLEIN, 2015).

Os sistemas ERPs normalmente são contratados em forma de um pacote de software e divididos em módulos, que são definidos de acordo com as demandas da empresa que os contrata e que se comunicam entre si, atualizando um mesmo banco de dados. Assim as informações inseridas em um módulo são automaticamente disponibilizadas para os demais módulos conectados a elas.

No mercado atualmente existe uma gama extensa de opções de ERPs disponíveis, porém muitas dessas soluções possuem uma limitação de módulos e trabalham com um tipo de segmento específico. Uma solução consolidada hoje no mercado, que possui uma lista extensa de módulos e atende empresas de diversos tamanhos e segmentos, nacionais e internacionais, é a TOTVS *Backoffice* - Linha Protheus. Um sistema ERP desenvolvido pela empresa Totvs com mais de 10 mil clientes e 200 milhões de acessos em módulos do sistema por ano (TOTVS, 2022a).

Apesar de ser um sistema robusto e de grande capacidade, o sistema Protheus apresenta algumas desvantagens. A primeira está relacionada a limitação de acesso por gestão de licenças, na qual cada usuário logado no sistema consome uma licença de acesso e estas licenças possuem um alto custo de adesão e mensalidade. Outra desvantagem é a interface complexa e pouco amigável, que requer treinamento de uso e alguns conhecimentos técnicos.

Para algumas situações estes pontos se tornam um gargalo no sistema e processo de quem usa o ERP, como no caso do módulo de Estoque e Custo que possui uma solução eficiente para requisições feitas ao armazém, automatizada e integrada diretamente ao módulo responsável pela compra de materiais. Porém, essas requisições normalmente são feitas por muitos funcionários diferentes. Por exemplo, no caso de materiais de consumo interno, como canetas, computadores e materiais de limpeza, que pode ser feita por todos os funcionários da empresa.

A realização via sistema necessitaria que fosse contratada uma licença por usuário ou até mesmo licenças ilimitadas, o que resultaria em um custo de aquisição e mensal bem alto. Além disso, seria preciso realizar treinamento e capacitação de todos os funcionários da empresa para a utilização do sistema, que não é muito intuitivo nem amigável, situação que normalmente envolve pessoas com diferentes níveis de escolaridade e com resistência a novas tecnologias. Outra solução, seria centralizar as requisições em algumas pessoas, porém isso causaria uma sobrecarga nesses funcionários, atrapalhando as suas atividades principais.

Para [Martelli e Dandaro \(2015\)](#) todas as empresas, independente do segmento, devem realizar a gestão e controle de estoque como fator de grande importância para a saúde financeira e competitividade no mercado. O estoque de uma empresa, por menor que seja, se mal administrado pode gerar trabalho excessivo, atraso de processos e afetar diretamente o caixa da empresa. Isso faz com que os módulos de gerenciamento de estoque seja um dos mais populares entre as demandas de ERP.

O armazém é um dos setores essenciais em boa parte das empresas, que exercem muitas funções de controle e manutenção de estoque que envolvem a empresa e os processos como um todo. Entre essas atividades o fornecimento de materiais de uso e consumo interno é uma das atividades que gera uma alta demanda, além de envolver diversos atores.

Diante deste cenário se torna essencial que essas requisições sejam realizadas de forma mais automatizada, fazendo com que a inclusão de informações manualmente seja a mínima possível. Gerando assim, uma otimização de tempo a todos os envolvidos, evitando prejuízos econômicos e atraso em processos.

Uma empresa de médio a grande porte que utiliza o sistema Protheus *Backoffice*, porém não utiliza a automatização de requisição ao armazém enfrenta diversas complicações. A iniciar-se pelo número de possíveis solicitantes, que podem se resumir a todos os funcionários da empresa na maioria dos casos.

Neste cenário, para que os funcionários possam fazer uma requisição de material é necessário o preenchimento de formulários manuais ou virtuais, ou até mesmo uma requisição informal, o que dificulta acesso a informações como lista de produtos, quantidades disponíveis e acompanhamento do processo de compra de um material solicitado.

Para o setor do armazém, este cenário se torna ainda pior, pois pode gerar filas no setor e acúmulo de papéis. Ao receber uma requisição é necessária a inclusão manual da movimentação de estoque ou inclusão de Solicitação de Compra para produtos sem estoque, o que pode envolver também inclusão manual no sistema pelo setor de compras. Além disso, é necessário fazer um controle interno da autorização de gestores e registro de entregas.

O fato deste processo possuir as informações preenchidas totalmente de forma manual, sem nenhuma automatização entre as rotinas, gera retrabalho dos funcionários, falta de informação dos solicitantes e torna o custo do processo mais elevado do que deveria.

Como o Protheus é um sistema amplo que atende a diversos segmentos ele disponibiliza várias formas de personalização interna e também algumas maneiras de comunicação com outros sistemas, como por exemplo, através de RESTful API. Por meio disto, este trabalho propõe uma solução para o gargalo de requisições ao armazém de forma que não seja necessária a contratação de muitas licenças de acesso ao Protheus, através do desenvolvimento de um sistema web de código aberto, com interface amigável e intuitiva, baseada em sites de compras, para a realização das requisições ao armazém e acompanhamento das mesmas pelos solicitantes.

A aplicação web utilizará de algumas das personalizações disponíveis e será conectada ao Protheus e ao seu banco de dados via API REST, consumindo apenas uma licença especial direcionada a este tipo de serviço. Isso faz com que a empresa possa aderir a solução de requisição ao armazém disponibilizada pelo Protheus, porém sem a necessidade de contratar inúmeras licenças de acesso, passando a contratar apenas uma licença de aplicação REST, o que irá diminuir o custo de forma significativa comparada a contratação de licenças de acesso para todos os funcionários. Além de fornecer aos funcionários uma plataforma familiar e similar a ideia dos *marketplaces* e seus carrinhos de compras, o que facilita para os funcionários que não possuem familiaridade com muitas tecnologias diferentes.

1.1 Objetivos

O objetivo principal deste trabalho é desenvolver uma aplicação *web* para o processo de automatização de requisições ao armazém de empresas que fazem uso do sistema ERP Protheus *Backoffice*, em que a requisição possa ser feita através da aplicação *web* e ser atendida pelo armazém através do Protheus. No qual a inserção de informação no sistema e movimentação de estoque será a mais automatizada possível. Além de não precisar da contratação de muitas licenças de acesso, tornando a implementação de baixo custo e com interface amigável e intuitiva ao público diversificado.

Para isso a aplicação deve fornecer acesso a todos os funcionários registrados e ativos da empresa, exibir todas as solicitações feitas por cada usuário e as respectivas informações necessárias para acompanhamento da requisição. A aplicação deve ainda permitir que sejam incluídas novas requisições, além de editar e/ou excluir as já existentes.

1.2 Metodologia

A metodologia de desenvolvimento para execução deste sistema é definida da seguinte maneira:

- Identificar as melhorias que o processo de requisição ao armazém automatizado pode trazer para os funcionários do armazém, os funcionários solicitantes de materiais e a empresa como um todo.
- Realizar o levantamento de requisitos baseado no uso padrão da solução de requisição ao armazém do ERP Protheus.
- Desenvolvimento das APIs necessárias para a comunicação com o sistema.
- Desenvolvimento da aplicação web com interface intuitiva e funcionalidades objetivas da solução, utilizando as APIs desenvolvidas e as informações enviadas através delas.
- Validação das comunicações e usabilidade.
- Análise da aplicação e do atendimento das requisições.

1.3 Organização do trabalho

O restante deste trabalho é organizado como se segue. O [Capítulo 2](#) apresenta a Fundamentação Teórica no qual o trabalho foi baseado e os trabalhos relacionados, o [Capítulo 3](#) apresenta o Desenvolvimento, tudo que foi utilizado e realizado durante o andamento do trabalho, o [Capítulo 4](#) apresenta todos os resultados obtidos pelo trabalho e por fim o [Capítulo 5](#), que realiza a análise dos objetivos alcançados e pontos de melhoria.

2 Fundamentação Teórica

Este capítulo apresenta uma revisão teórica dos conceitos e ferramentas utilizadas, bem como trabalhos correlatos.

2.1 ERP TOTVS *Backoffice* - Linha Protheus

O Protheus é um ERP robusto, capaz de atender empresas de qualquer porte, e atende a 12 seguimentos diferentes: Agronegócio, Atacado e Distribuição, Indústria, Logística, Serviços, Construção e Projetos, Saúde, Varejo, Educacional, Jurídico, Hospitalidade e Finanças. (TOTVS, 2022b)

O sistema possui integração com soluções especialistas dos mais diversos setores e também soluções com tecnologia móvel desenvolvidas especialmente para a ferramenta. Além disso, é um sistema considerado escalável com configurações e integrações personalizáveis, oferecendo uma ferramenta online de ajuda com uma documentação extensa e completa, fazendo com que seja um sistema que se adapte a todos os tamanhos e especificidade das empresas.

A solução oferece uma lista ampla de módulos que trabalham de forma integrada e tem o objetivo de simplificar rotinas de trabalho, reduzir erros operacionais, digitação manual e retrabalho em todas as tarefas. E ainda ajuda as empresas a padronizar processos e aderir a melhores práticas de negócios.

Para as personalizações e administração em geral, o sistema possui um módulo especial chamado Configurador, por onde os usuários administradores podem manipular e criar informações no sistema. Além deste módulo, a TOTVS permite personalizações mais complexas no sistema, através de Pontos de Entradas e também disponibiliza formas de integração a outros sistemas através de Web Services e APIs.

2.1.1 Módulo Estoque e Custos

Entre os principais módulos oferecidos está o de Estoque e Custos, que permite a empresa controlar o seu estoque de forma rigorosa, definir níveis mínimos de estoque, realizar pedidos de compra automáticos, entre muitas outras soluções voltadas para área.

Uma dessas soluções é a Solicitação ao Armazém, que permite os usuários do sistema solicitar produtos ao armazém. Feitas essas requisições, o funcionário responsável pelo armazém gera a requisição, que faz a análise do estoque e pode gerar solicitações de compras automáticas de acordo com os parâmetros definidos pelo usuário. Além disso, ao

entregar o produto, o funcionário do armazém já realiza a baixa direta do estoque, sem necessidade de inserir uma movimentação de estoque manual. Para saber mais detalhes técnicos, a Totvs oferece uma guia de referência sobre essas rotinas, que pode ser encontrada [aqui](#).

O fluxo padrão dessa solução pode ser observado através da [Figura 1](#), na qual a Solicitante realiza *login* no Protheus, acessa a rotina de Solicitar e inclui uma Requisição. E o Atendente, acessa a rotina de Gerar Pré-Requisição para atender a requisição, e posteriormente a rotina de Baixa de Requisição quando entrega o produto a Solicitante.

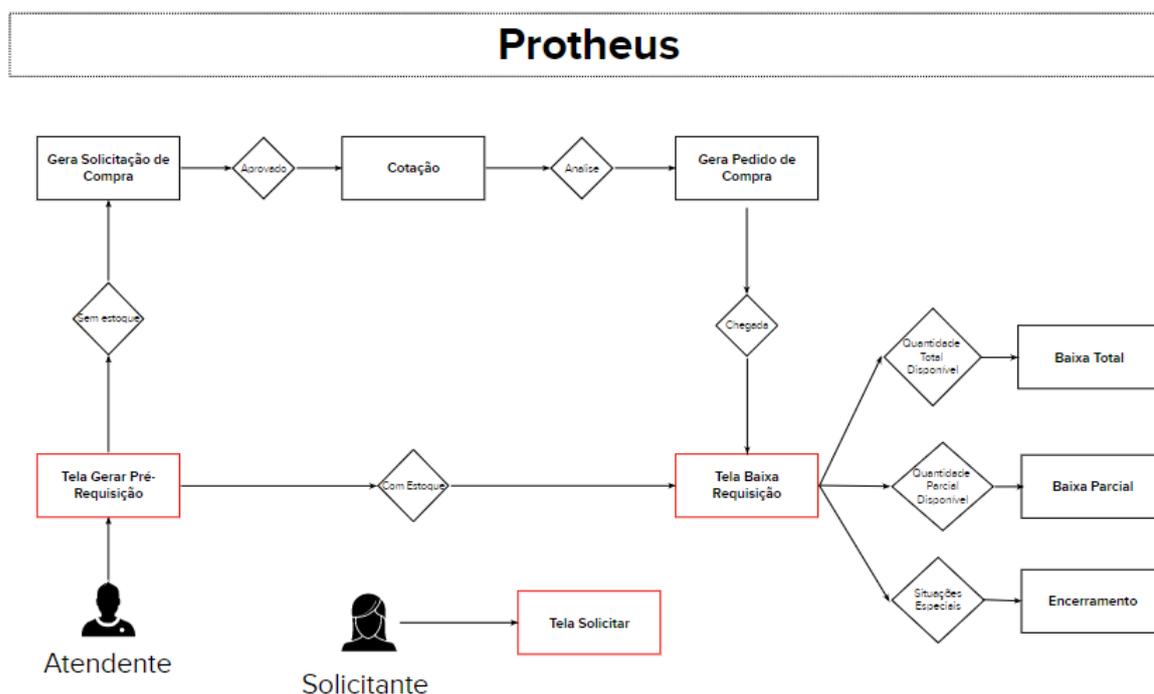


Figura 1 – Fluxo de Requisição ao Armazém Padrão do Protheus.

Fonte: elaborado pela autora.

2.2 API REST

A arquitetura *Representational State Transfer* (**REST**), na tradução, Transferência de Estado Representacional, foi proposta por ([FIELDING, 2000](#)) e vêm sendo adotada como o método preferido para conectar dois sistemas, em especial sistemas *webs*.

Ela permite o desenvolvimento de uma *Application Programming Interface* (**API**), que é uma interface de programação de aplicação, usada para implementar padrões e protocolos em uma aplicação, ela define um conjunto de normas que possibilita a comunicação entre dois sistemas diferentes.

A construção da API REST também conta com a utilização do formato de dados *JavaScript Object Notation* (JSON) para a compartilhamento dos dados. Este formato transfere os dados para um formato textual, que ao aderir a este formato passa a poder ser lido ou gerado por qualquer aplicação, independente da linguagem ou banco de dados utilizado.

A conexão e disponibilização de informações entre o Protheus e a aplicação desenvolvida é feita através de uma RESTful API. Na qual, o serviço REST irá conectar o ERP Protheus à aplicação *web* desenvolvida e a API enviará os dados através do formato JSON utilizando os padrões e protocolos definidos.

2.3 Trabalhos Relacionados

Nenhum sistema foi encontrado com o mesmo objetivo de ser um sistema *web* de gerencia de solicitação ao armazém voltado para a integração ao Protheus. Então, foram selecionados alguns trabalhos voltados ao estudo e análise do ERP na gestão de estoques.

2.3.1 A importância do uso de *software* ERP como ferramenta para um eficiente controle de estoque

O estudo PADULA et al. (2021) relata um estudo de caso de uma empresa de prestação de serviços e venda de peças de reposição, com área de atuação em poços artesianos e reforma de bombas submersas.

O artigo tem como objetivo demonstrar que, apesar da prefeitura municipal fornecer um *software* que gera as notas fiscais de serviços e, embora tal sistema seja gratuito, ainda assim há necessidade de se adquirir um *software* mais robusto e do tipo ERP para controlar os estoques.

Do ponto de vista da atividade principal da empresa, prestação de serviços, não necessitaria de um *software* de gestão integrada para gerir apenas a emissão das notas fiscais. Entretanto, para auxiliar a atividade principal, outras atividades são executadas em paralelo, como controle de contas a receber e pagar, controle de entradas e saídas de materiais e controle de estoque.

Para realizar as assistências são utilizadas peças de reposição que são fabricadas pelas indústrias e compradas pela empresa, visto que pela quantidade comprada as empresas de assistência possuem uma vantagem na compra.

Além disso, na prestação de serviços são exigidos o cumprimento de vários procedimentos relacionados à manutenção dos equipamentos utilizados, à segurança dos colaboradores no manuseio dos equipamentos que são testados, utilizando energia elétrica, estática, e os padrões estabelecidos pelos fabricantes das bombas submersas.

Essas situações fazem com que o cliente exija uma documentação com estas atividades, o detalhamento das peças utilizadas e os serviços executados. O que exige que a empresa precise realizar o controle da entrada das peças de reposição e posteriormente, a saída com a emissão da nota fiscal a cada equipamento reformado.

No âmbito fiscal, a empresa está enquadrada no Simples Nacional, no qual o imposto de Imposto Sobre Serviços (ISS) é pago juntamente aos demais impostos que a empresa faz o recolhimento. Sendo o ISS um imposto de competência municipal, o município oferece um *software* que permite a emissão das notas fiscais e que faça o cálculo do percentual do ISS sobre o valor do serviço prestado, posteriormente apurado junto à guia de recolhimento do Simples Nacional.

Porém este *software* não possui suporte para a das notas fiscais das peças de reposição para a reforma dos equipamentos, essas peças possuem outros tipos de impostos, como Imposto sobre Produtos Industrializados (IPI), Imposto sobre Circulação de Mercadorias e Prestações de Serviços (ICMS) e em alguns casos, Imposto Sobre Serviços (ST).

Sendo assim, a empresa fez a aquisição de um segundo *software*, que permite um controle melhor em todas as atividades, entradas, saídas, controles financeiros, controle de estoque e suporte as notas fiscais das peças de reposição. Porém o novo *software* não possuía integração com o *software* de emissão de notas fiscais de serviço.

O trabalho então recomendou a integração dos *softwares*, que teve um adicional mensal no custo do *software* contratado em 83% do valor inicial. Entretanto, todo o processo passou a ser centralizado e os benefícios de ter um ERP, mesmo que simplificado como o que foi contratado, foram identificados e o custo mensal foi considerado satisfatório para a empresa.

2.3.2 Planejamento e Controle de Estoque nas Organizações.

O artigo [Martelli e Dandaro \(2015\)](#) tem o objetivo de analisar os estoques, seus tipos e as ferramentas disponíveis de gestão com foco o planejamento e controle dos materiais, de forma que atenda a demanda de uma empresa independentemente do segmento em que atua, sem desperdícios no suprimento e com agregação de valor aos investimentos organizacionais.

O artigo é direcionado a ambientes industriais, porém apresenta conceitos que também podem ser aplicados em outros segmentos. Traz a importância aos sistemas de controle de gestão de estoque como o plano mestre de produção, *Manufacturing Resource Planning I (MRP I)*, *Manufacturing Resource Planning II (MRP II)* e o ERP.

Os autores afirmam que a grande vantagem do sistema ERP é a habilidade e rapidez que o sistema possui quando alimentado de informações lançadas nele até o momento da chegada aos setores envolvidos na solicitação do serviço, tudo em tempo

real, interagindo com todos os aplicativos e eliminando-se a redigitação de dados, o que assegura a integridade das informações obtidas.

Por fim, os autores concluem que o MRP II juntamente com a ERP, podem ser ferramentas essenciais para o controle dos materiais alinhados a necessidade da empresa como um todo, por meio de informações compartilhadas e sincronizadas pelos diversos departamentos da organização.

2.3.3 ERP - Módulo ESTOQUE/CUSTOS, PCP e os problemas burocráticos

O artigo [Silva e Najm \(2014\)](#) aborda os problemas que envolvem a burocracia de uma empresa e seu ERP. Situações em que não se consegue adequar o *software* a empresa, muitas vezes porque a empresa está com seu processo desestruturado.

O artigo apresenta um estudo de caso de uma empresa brasileira que atua no segmento industrial do mercado metalúrgico. A empresa se encontrava em processo de implantação de MRP dentro da plataforma do ERP Protheus, para administrar e gerenciar o estoque, planejamento e cadeia de custos.

elaborado pela autora colocam em foco os módulos PCP e ESTOQUE/CUSTOS, e quais as consequências nos chamados “furos de estoque” (que são divergência entre o saldo físico e do sistema), e por consequência a dificuldade de se controlar o custo do estoque, e para se fazer uma tomada de decisão a nível de planejamento, podendo levar a prejuízos financeiros ou falta dos materiais no momento que o mesmo for solicitado.

Os principais motivos apontados pelos autores para a divergência no saldo de estoque físico e sistêmico são:

- As notas fiscais não são imediatamente lançadas no sistema, no ato do recebimento, o que não atualiza o estoque no sistema. Isso ocorre devido a erros na criação e/ou falta do pedido de compra, ou erro na nota fiscal do fornecedor. Problemas esses que não ocorreriam, se fosse cumprido o procedimento de não receber materiais sem o lançamento da nota fiscal no sistema, mas a própria gerência pressionada pela falta do material, para evitar um impacto negativo na produção, por isso burla esse procedimento (ou burocracia) e os materiais são recebidos sem o saldo no sistema existir.
- O método de requisição de materiais de consumo não é sistematizado desde o início, ou seja, são feitas em blocos de papel, onde são baixados no sistema posteriormente e pode ocorrer a falha humana de ser esquecido de baixar, ou não ter saldo para baixar a mesma devido ao problema citado no parágrafo anterior.
- O método de baixa de materiais de industrialização é falho, pois para cada projeto e ordem de produção, é feita uma lista de necessidades de matéria-prima, que já é

pré-definida pela engenharia de projetos, que, se for mudado algum componente, o acfPCP tem que fazer ajuste manual, para que o armazém consiga fazer a baixa do mesmo no sistema, o que esbarra em muitos erros humanos, pois a quantidade de alterações é grande e exige uma enorme concentração e dedicação para cada caso.

- Os produtos acabados, em sua grande maioria não ficam na expedição com o saldo do sistema correto, pois a burocracia do método de apontamento de produção é burlado até mesmo pela gerência da empresa e o sistema existente de apontamento não suporta a realidade da empresa, com isso os materiais acabados são liberados para a expedição sem o devido saldo atualizado, o que afeta diretamente o faturamento, que não é feito sem o saldo no sistema.

Como conclusão os autores trazem alguns pontos: a falta de atualização do sistema que pode estar deixando os processos do sistema muito burocráticos; processos da empresa desatualizado, cultura ultrapassada e/ou problemas de gestão; e falha na implantação do sistema, na escolha do mesmo, ou na aderência entre os processos da empresa e o configurado no ERP.

2.3.4 Conclusão

Diante destes trabalhos é possível concluir que a contratação de um software ERP se torna um requisito essencial para as empresas, trazendo muitos benefícios de controle e processos à empresa, e conseqüentemente, benefícios financeiros. Entretanto, também foi possível observar que a obtenção dos benefícios de um software ERP também depende do seu uso adequado e da inserção correta dos dados. O que valoriza a importância de definição dos processos de acordo com a atualidade do mercado e automatização desses processos, visando diminuir a burocracia e manter a integridade e veracidade dos dados.

Além disso, é possível identificar como a contratação de um software robusto traz um custo benefício maior para a empresa, já que a integração entre diferentes *softwares* pode ter um custo elevado.

Em especial, vale ressaltar o último artigo abordado, ERP - Módulo Estoque/Custos, PCP e os Problemas Burocráticos. Ele aborda o ERP Protheus e o foco deste trabalho. O estudo de caso deste artigo mostra uma situação um pouco semelhante ao estudo de caso apresentado neste trabalho mais adiante, em especial os dois primeiros tópicos apresentados pelos autores como principais motivos para a divergência no saldo de estoque. A falta de automatização no processo de requisição ao armazém (retiradas do armazém) e solicitações de compras causa um grande transtorno e abre margem para diversos outros erros no processo. A solução desenvolvida neste trabalho juntamente a uma mudança cultural da empresa e dos setores de compras, estoque e PCP solucionariam grande parte desses problemas de estoque divergentes.

3 Desenvolvimento

Este capítulo descreve o desenvolvimento do trabalho, as ferramentas utilizadas, um estudo de caso, a modelagem utilizada e o que foi necessário na implementação.

3.1 Ferramentas e tecnologias

Esta seção lista as principais linguagens de programação, ferramentas e tecnologias utilizadas durante o desenvolvimento deste trabalho.

3.1.1 Linguagens

As linguagens de programação utilizadas foram as seguintes:

- HTML 5: linguagem padrão para o acesso e exibição de páginas Web;
- CSS: folhas de estilo em cascata, responsáveis pela personalização do conteúdo das páginas *Web*, como fonte das letras, tamanho, cores, bordas e muito mais;
- PHP 7.4.19: linguagem de *script* voltada para o lado do servidor e para a manipulação de dados;
- *Javascript*: linguagem de *script* que permite a criação de páginas atrativas e interativas aos usuários, possibilitando realizar muitas tarefas importantes e enriquece de forma notória uma página Web;
- *Advpl*: linguagem de programação criada pela própria TOTVS para personalizações e integrações mais complexas no Protheus. Possui muitas funções disponibilizadas para facilitar o desenvolvimento e conectar às rotinas diretamente, sem a necessidade de acessar aos códigos fontes do sistema Protheus.

3.1.2 *Laravel*

Laravel é um *framework* de aplicação *web* para PHP, com código aberto lançado em 2011. Fornece uma estrutura e um ponto de partida para criar a aplicação, é considerado um dos melhores para o desenvolvimento *web* e *full-stack*, no qual o desenvolvedor realiza o trabalho tanto do *back-end* quanto do *front-end*. Foi escolhido principalmente porque possui uma estrutura escalável e facilitadora da conexão com as APIs. A versão do *Laravel* utilizada é a 8.83.25. (LARAVEL, 2022)

3.1.3 *Bootstrap*

Lançado em 2011 pelo *designer* do *Twitter*, o *Bootstrap* é um dos *frameworks* mais conhecidos do mundo, principalmente no que se trata de estilo e responsividade. Ele fornece um kit de ferramentas de *front-end* poderoso, extensível e repleto de recursos. Com ele é possível utilizar sistemas e componentes de grade pré-construídos, que facilitam a construção do *layout* da aplicação, sem necessariamente criar um novo componente no CSS, além dos plugins em JavaScript. ([BOOTSTRAP, 2022](#))

3.1.4 *JQuery*

JQuery é uma biblioteca *JavaScript* rápida, pequena e rica em recursos. Ela facilita o uso do *JavaScript* e também é usada nos *plugins* da versão do *Bootstrap* usada neste trabalho. ([JQUERY, 2022](#))

3.1.5 *AdminLTE*

O *AdminLTE* é um modelo de painel de administração de código aberto e tema do painel de controle. Construído sobre o *Bootstrap*, o *AdminLTE* fornece uma variedade de componentes responsivos, reutilizáveis e comumente usados. ([ADMINLTE, 2022](#))

3.1.6 *SQL Server*

O SGBD utilizado pelo Protheus que foi usado como base do desenvolvimento é *Microsoft SQL Server*. A linguagem utilizada neste SGBD é a *Transact-SQL* que é baseada na linguagem SQL. Sendo assim, toda as APIs foram desenvolvidas com base nesta linguagem de consulta ao banco de dados. ([MICROSOFT, 2022](#))

3.1.7 *GitHub*

Para controle de versionamento do projeto foi utilizado o GitHub. Uma plataforma que permite criar repositórios com códigos fonte e também possibilita os usuários a acessar projetos *open-source*, colaborar, criar chamados e fóruns de dúvidas. Amplamente utilizado pelos desenvolvedores do mundo. Como o objetivo deste projeto é ser um desenvolvimento *open-source*, o *GitHub* se tornou a plataforma perfeita para disponibilizar todos os fontes deste trabalho. ([GIT, 2022](#))

3.1.8 *Mural*

Mural é uma ferramenta que pode ser utilizada em vários segmentos e com vários objetivos. É uma tela em branco em que se pode inserir diversos tipos de elementos, no qual várias pessoas podem acessar, visualizar e colaborar com o mesmo projeto. Utilizada por

designers, equipes de recursos humano, analistas de sistemas, entre muitos outros públicos. Neste trabalho ela foi utilizada para desenhar os protótipos das telas e compartilhar com funcionários colaboradores do estudo de caso. (MURAL, 2022)

3.1.9 Astah UML

O *Astah UML* é um *software* de modelagem de para modelagem *Unified Modeling Language (UML)*, ou Linguagem de Modelagem Unificada. O *software* possui uma versão gratuita que atende muitas funcionalidades, em especial a utilizada neste trabalho, a criação de diagramas. Neste trabalho o *software* foi utilizado para gerar todos os diagramas e modelagens. (ASTAH, 2022)

3.1.10 Postman

Postman é uma plataforma para construir e testar APIs. O *Postman* simplifica cada etapa do ciclo de vida da API e agiliza a colaboração para que você possa criar APIs melhores e mais rapidamente. Ele possui diversas funcionalidades, entre elas, a mais usada é o teste de APIs, sejam eles automatizados ou não. (POSTMAN, 2022)

3.1.11 VS Code

Visual Studio Code, ou *VS Code*, é um editor de código-fonte desenvolvido pela *Microsoft* que inclui suporte para depuração, controle de versionamento *Git* incorporado, entre vários outros *plugins* e extensões que podem incrementar a área de desenvolvimento do programador. (VISUAL STUDIO CODE, 2022)

3.2 Estudo de Caso

A fim de tornar o trabalho mais assertivo possível, foi selecionada uma empresa anônima que utiliza o ERP Protheus e atualmente realiza as requisições ao Armazém inserindo todas as informações no sistema de forma manual, com o objetivo de implantar e testar um protótipo. A empresa selecionada não adota a solução automática de requisições ao armazém mencionada na [subseção 2.1.1](#), pois para isso seria necessário que todos os funcionários que realizam requisição ao armazém possuíssem licença de acesso ao Protheus, o que resultaria em um custo muito alto para a grande maioria utilizar apenas uma rotina. Sendo assim, a empresa realiza o processo de forma manual, e inclui as movimentações de estoque diretamente na rotina de Movimentações Múltiplas. A [Figura 2](#) apresenta o fluxo deste processo manual de requisições da empresa.

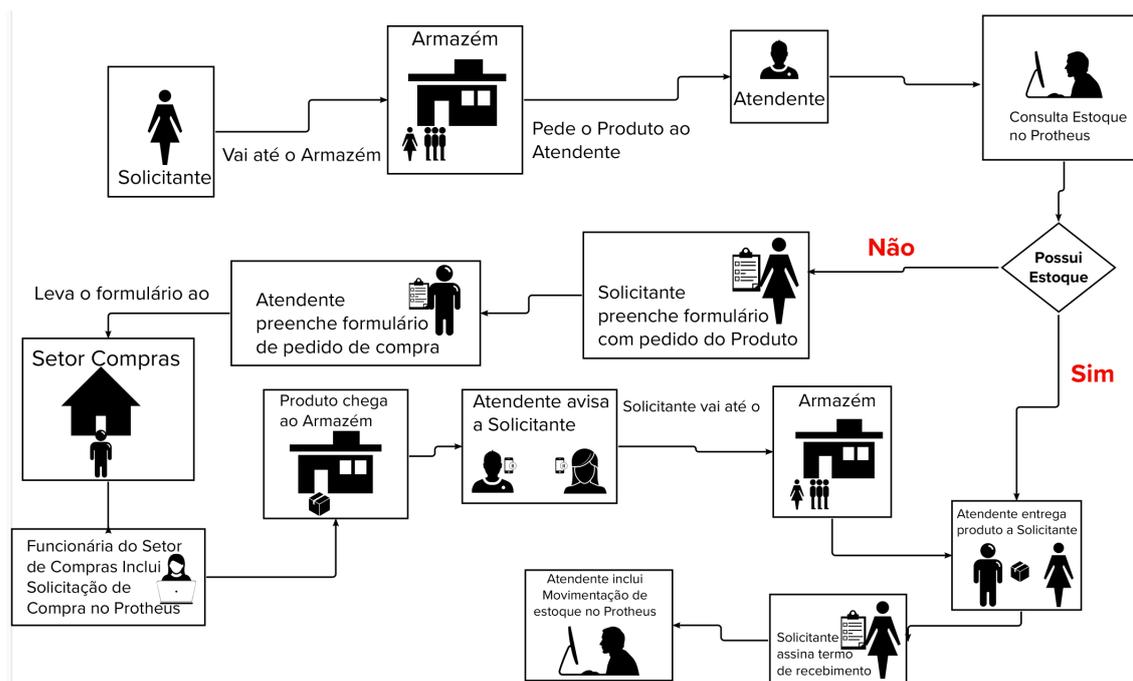


Figura 2 – Fluxo do processo de requisição ao armazém do Estudo de Caso.

Fonte: elaborado pela autora.

Diante do processo apresentado pela empresa, alguns pontos foram identificados:

- Acúmulo de papel: A solicitação do requisitante é feita através de um papel, a solicitação do funcionário do armazém ao departamento de compras é feita através de papel e a entrega do produto ao requisitante também é feita através de assinatura em um papel.
- Demora no processo: O requisitante faz a sua requisição manual, o funcionário do armazém busca pelo produto no sistema, nem sempre no mesmo momento e assim define se a entrega será realizada naquele momento ou se será necessário realizar um processo de compra.
- Filas no armazém: Como o processo é lento e sobrecarrega os funcionários do armazém, acaba gerando filas em alguns horários.
- Falta de transparência: O requisitante não tem acesso aos produtos disponíveis no sistema, nem aos estoques. Nos casos em que é necessária a solicitação de compra, o requisitante não consegue acompanhar o processo nem ter uma previsão de entrega do seu produto.
- Inclusão manual no sistema: Ao final do processo e entrega do produto é feita inclusão manual da movimentação no sistema pelo funcionário do armazém. Em casos de necessidade de solicitação de compra, a solicitação também é incluída manualmente pelo funcionário do departamento de compras.

- Informações inconsistentes no sistema: A movimentação só é incluída no sistema quando o produto já foi entregue, logo a quantidade do produto permanece errada até o momento da inclusão. Em caso de solicitação de compra, quando a nota de entrada é realizada no sistema o estoque do produto fica com a quantidade disponível, quando na verdade já está reservada.

Baseado nesses pontos e no funcionamento da solução de requisição ao armazém padrão do Protheus, foi levantado os seguintes pontos:

- O protótipo deve ser capaz de permitir que o requisitante visualize todas as suas requisições;
- O protótipo deve ser capaz de permitir que o requisitante consiga acompanhando o status do processo em que se encontra sua requisição;
- O protótipo deve ser capaz de permitir que o requisitante inclua uma nova requisição;
- O protótipo deve ser capaz de disponibilizar ao requisitante os dados de todos os produtos necessários à sua solicitação;

A fim de exemplificar o pontos levantados foi criado o diagrama de caso de uso da [Figura 3](#), que exemplifica a principal função da aplicação: visualizar e incluir requisições.

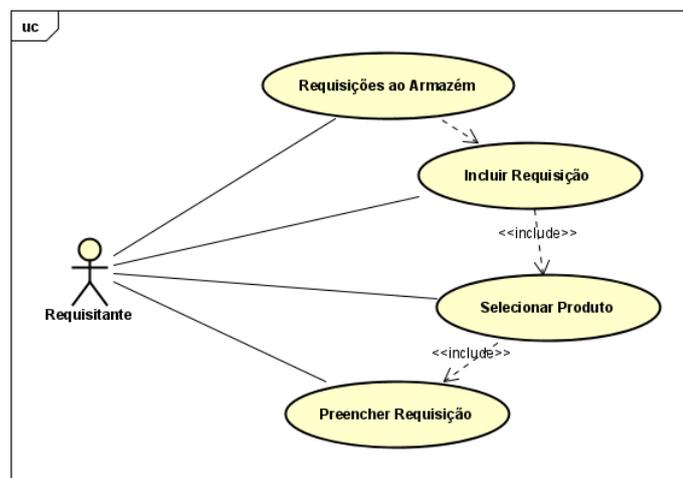


Figura 3 – Caso de uso do protótipo.

Fonte: elaborado pela autora.

Baseado nas informações e no diagrama apresentado, foram criadas as telas de prototipação das principais páginas. O formato foi desenhado pensando em fazer a aplicação comparada a compra de um produto em um site de *e-commerce*, com o objetivo de trazer familiaridade do usuário em relação ao processo.

Com isso o fluxo seria parecido ao de um *e-commerce* básico, como apresentado na Figura 4.

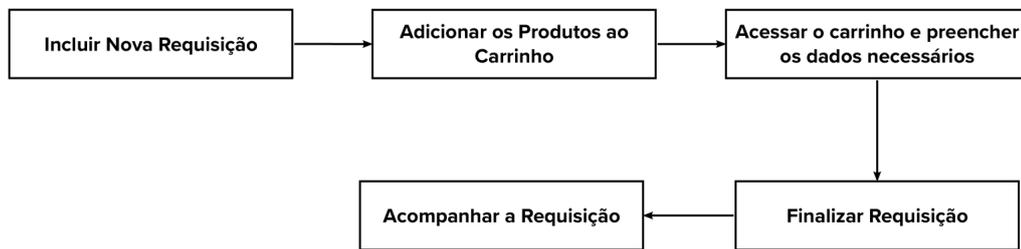


Figura 4 – Fluxo do Caso de Uso do Protótipo.

Fonte: elaborado pela autora.

Na Figura 5, é apresentada a página inicial do protótipo, considerando que o usuário já está logado. Nela pode-se observar na barra lateral o espaço para logo da empresa e o menu "Requisição ao Armazém". Na barra superior, em cinza, está o nome do sistema e o ícone de usuário, por onde o usuário pode clicar para sair do sistema.



Figura 5 – Protótipo da página inicial.

Fonte: elaborado pela autora.

Ao clicar no menu "Requisição ao Armazém", o usuário é transferido para a página

da Figura 6, a página principal. Nesta página é possível observar que na barra superior foi incluído um ícone de carrinho por onde é possível acessar o carrinho de requisições. No conteúdo está um botão para incluir uma nova requisição e logo abaixo um exemplo de visualização de uma requisição. Ao final da página, estão os botões de paginação para mudanças de páginas no caso em que as requisições ultrapassem o espaço definido. Ainda nesta página, é importante ressaltar que na visualização da requisição estão presentes, além das informações da requisição e seus itens, também os ícones de edição, exclusão e visualização de mais detalhes do status.

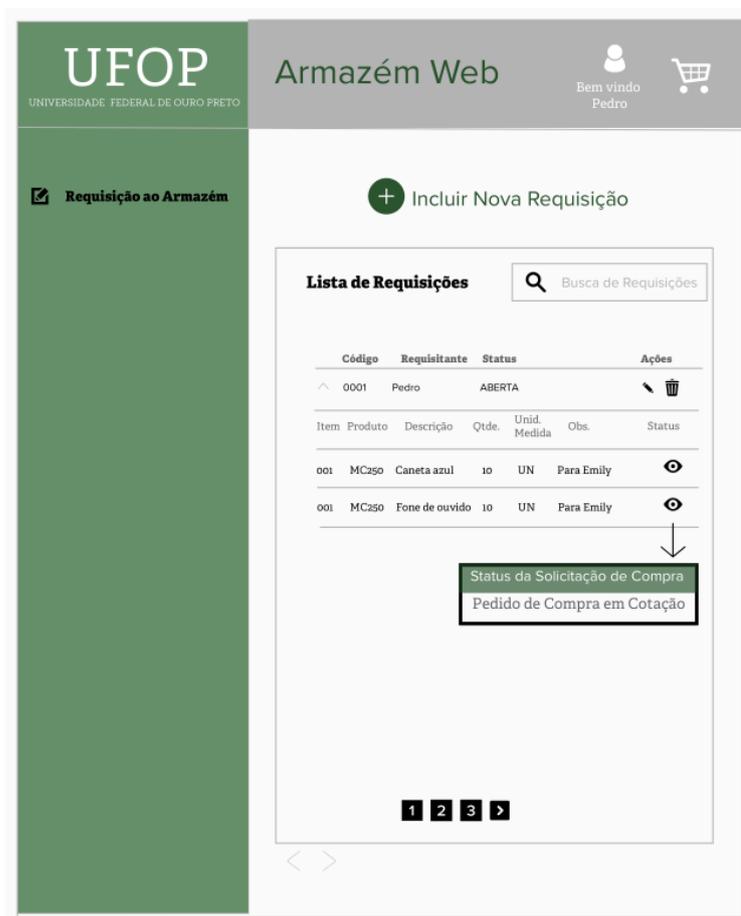


Figura 6 – Protótipo da página principal.

Fonte: elaborado pela autora.

A Figura 7, apresenta a página exibida após selecionar o botão "Incluir Nova Requisição", a página de produtos. Na qual o usuário pode selecionar todos os produtos necessários ao seu carrinho de requisições, filtrar a listagem de produtos por Tipos ou Grupo, ou pesquisar por um produto específico. Além de ter a opção de apenas adicionar os produtos selecionados ao carrinho e voltar a página inicial, ou já finalizar a solicitação e ir para a página de carrinhos.

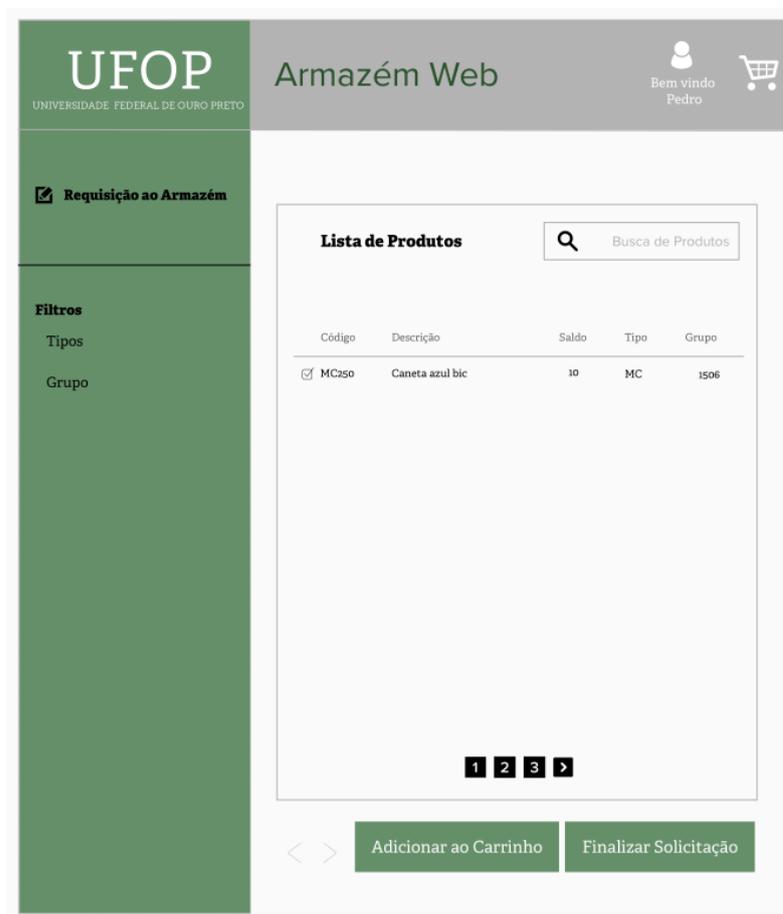


Figura 7 – Protótipo da página de produtos.

Fonte: elaborado pela autora.

Por fim, a [Figura 8](#) e [Figura 9](#) mostram a página do carrinho de requisições, em que a [Figura 8](#) representa a página no formato vazio para a situação que não foi selecionado nenhum produto e já apresenta um atalho com o botão de incluir nova solicitação, e a [Figura 9](#) a qual lista todos os produtos selecionados e armazenados no carrinho e os campos que precisam ser preenchidos para completar a requisição: centro de custo, quantidade e o campo opcional de observação.



Figura 8 – Protótipo da página vazia do carrinho de requisições.

Fonte: elaborado pela autora.



Figura 9 – Protótipo da página de preenchimento do carrinho de requisições.

Fonte: elaborado pela autora.

Esses protótipos foram apresentados a dois funcionários considerados chaves no processo a fim de verificar as dificuldades de usabilidade e as melhorias que o sistema poderia ter. Com isso, foram levantados os seguintes pontos:

- Melhorar a opção de voltar as páginas anteriores;
- Incluir uma página de informações com exemplos de uso do sistema;
- Incluir filtros diversificados na listagem de requisições;
- Incluir mais tipos de filtros na página de produtos;
- Incluir uma forma de visualizar mais detalhadamente as requisições, já que alguns textos podem ser muito grandes e também com outras informações menos relevantes;
- Simplificar a forma de incluir um novo produto;
- Incluir mais informações do produto no carrinho de solicitações.
- Padronizar as nomenclaturas como “Requisição” e seus derivados, pois a palavra solicitação foi confundida com o processo de apenas solicitação de compra.

3.3 Modelagem

A modelagem da aplicação foi baseada no protótipo criado e nos pontos de melhorias levantados durante o estudo de caso. Nessa seção serão apresentados os requisitos funcionais e não funcionais, diagrama de casos de uso, diagrama de atividade, diagrama de sequência e a estrutura do banco de dados envolvido na solução.

3.3.1 Requisitos

3.3.1.1 Requisitos Funcionais

Os Requisitos Funcionais foram gerados a partir do estudo de caso e do funcionamento padrão da solução de Solicitação ao Armazém do ERP Protheus, e são os seguintes:

- Permitir o cadastro de usuário que seja funcionário ativo da empresa;
- Permitir alteração da senha de usuário cadastrado e ativo;
- Permitir a visualização das requisições já feitas pelo usuário;
- Permitir ao requisitante o acompanhamento do processo da sua requisição;
- Permitir realizar um filtro detalhado das requisições já feitas pelo usuário;
- Permitir a inclusão de uma nova requisição;
- Permitir que o usuário tenha acesso a toda lista de produtos ativos cadastrada no sistema;
- Permitir que o usuário possa fazer um filtro detalhado da listagem de produtos;
- Permitir a alteração e exclusão de uma requisição, desde que esta ainda não tenha sido manipulada pelo Armazém;
- Disponibilizar formas de contato com o suporte;
- Disponibilizar um tutorial de uso da aplicação.

3.3.1.2 Requisitos Não Funcionais

O requisitos não funcionais foram implementados de forma a atender uma aplicação *open-source*, porém devem ser incrementados pelas empresas que adotarem a solução. A disponibilidade deve ser definida pela empresa, entretanto abaixo é feita uma sugestão. E a segurança deve ser melhorada pela empresa que o implementar, principalmente a respeito de *tokens* de segurança para acessar as APIs. Seguem os pontos aplicados no trabalho:

- Confiabilidade: O sistema deve mostrar resultados confiáveis e que correspondam com as informações solicitadas pelo usuário.
- Usabilidade: O sistema deve ser intuitivo e de aparência agradável, além de ser sucinto e claro com as regras de uso.
- Segurança: O sistema deve ter métodos de segurança para evitar que pessoas não autorizadas acessem as informações disponibilizadas na aplicação.
- Disponibilidade: O sistema deve estar disponível para qualquer usuário acessar de qualquer lugar que tenha internet e acesso à rede interna da empresa.

3.3.2 Casos de Uso

A [Figura 10](#) apresenta o diagrama de casos de uso da aplicação, no qual existe apenas um ator, o requisitante, que através de algumas verificações pode realizar o cadastro de *login* e recuperar senha. Além de realizar todas as outras funções do sistema, como visualizar as requisições e seus status, editar, excluir, incluir ou filtrar uma requisição; visualizar e filtrar a lista de produtos; adicionar um produto ao carrinho, acessar ao carrinho de requisições para finalizar a requisição.

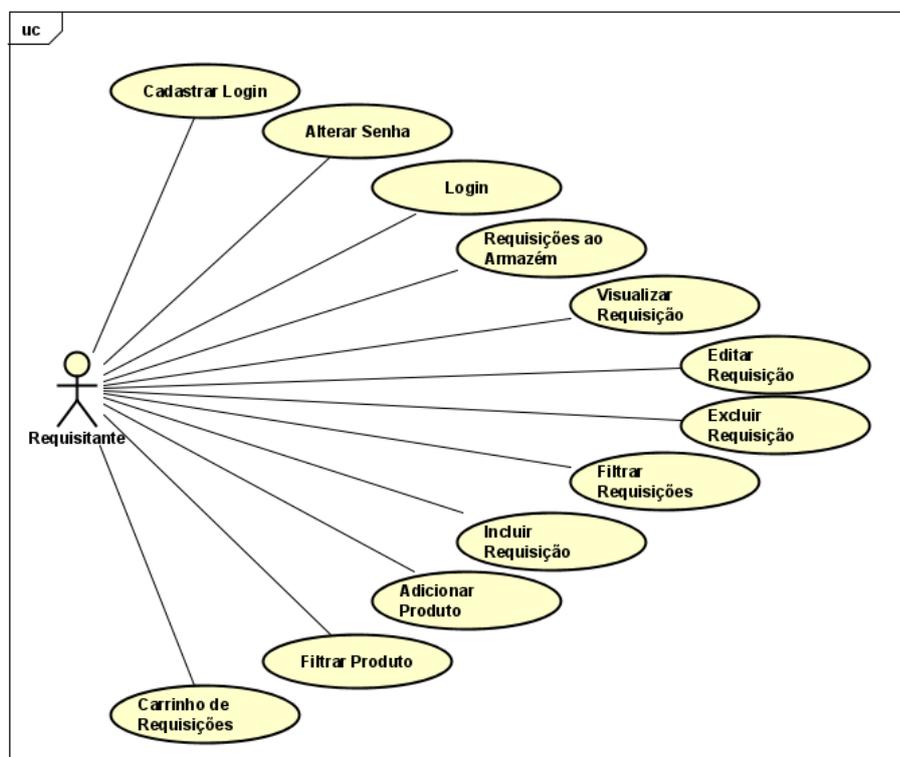


Figura 10 – Diagrama de Casos de Uso.

Fonte: elaborado pela autora.

3.3.3 Diagrama de Atividade

Para visualizar todas as dependências de editar uma requisição, foi feito o diagrama de atividade apresentado na [Figura 11](#). Os diagramas das outras atividades do sistema estão disponíveis no [Apêndice A](#).

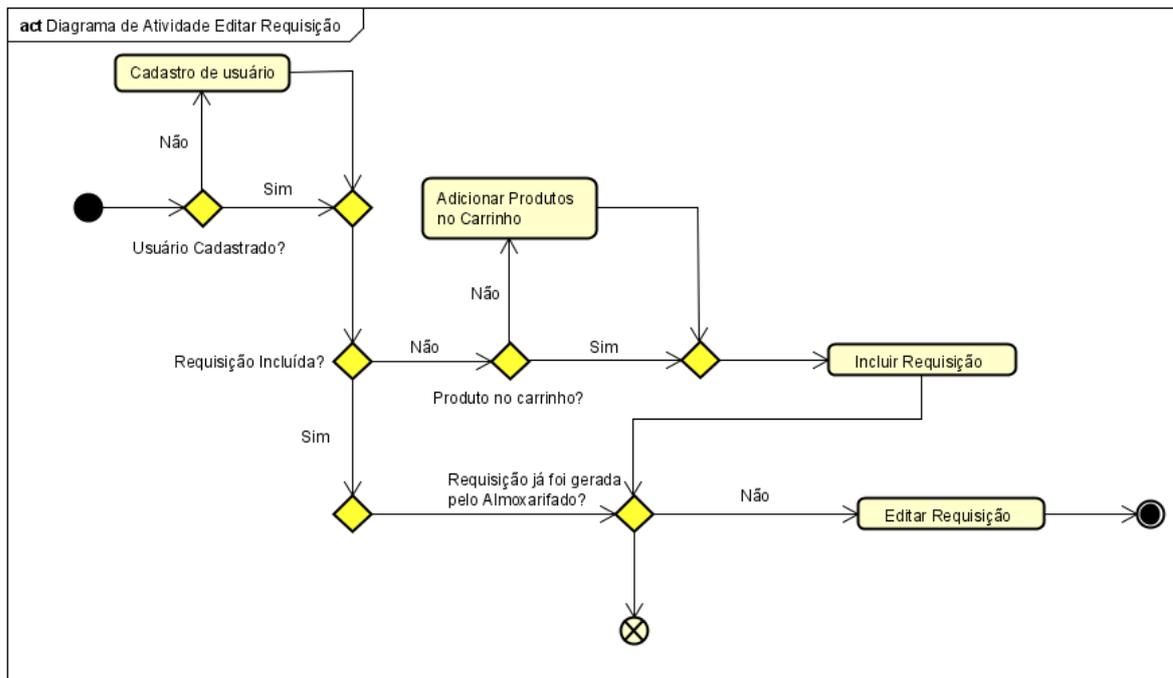


Figura 11 – Diagrama de Atividade referente a Edição de uma Requisição.

Fonte: elaborado pela autora.

3.3.4 Diagrama de Sequência

Para apresentar o passo a passo a ser seguido para a aplicação, foram criados os diagramas de sequência. Por eles é possível verificar todo o funcionamento e organização do sistema. Na [Figura 12](#) está o diagrama de sequência referente ao processo de incluir uma nova requisição. Nele é possível verificar o comportamento do sistema e os momentos de comunicação entre a aplicação e as APIs do Protheus. Os demais diagramas podem ser encontrados no [Apêndice A](#).

3.3.5 Banco de Dados

O banco de dados utilizado foi o banco do ERP Protheus. Este banco de dados contém todas as tabelas de todos os módulos do sistema e também as tabelas personalizadas. As tabelas do sistema possuem muitos campos, já que são criadas de forma genérica a atender vários segmentos. Por isso a título de informação serão apresentados neste trabalho apenas as tabelas e campos utilizados no desenvolvimento e aplicação.

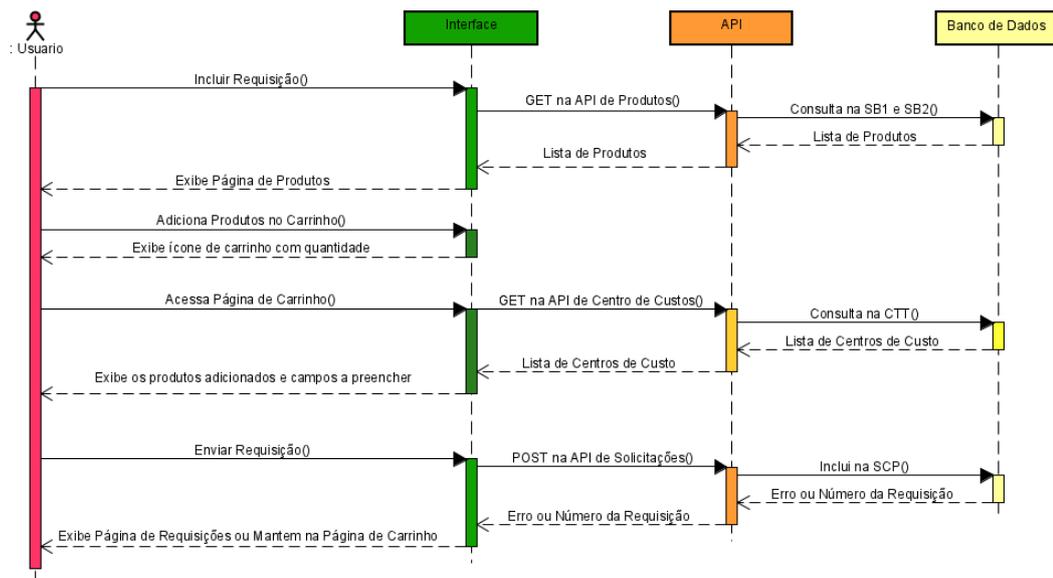


Figura 12 – Diagrama de Sequência de Incluir Nova Requisição.

Fonte: elaborado pela autora.

Para atender as necessidades da aplicação e manter todas as tabelas centralizadas no banco de dados do Protheus, foi necessária a criação de 2 campos personalizados e a criação de uma nova tabela para armazenamento dos usuários. Essas criações foram feitas através do módulo Configurador do Protheus, conforme orientação do [Apêndice B](#). Os campos personalizados possuem em sua nomenclatura os dígitos XX.

As tabelas utilizadas foram as seguintes:

- SRA: Funcionários
- CTT: Centro de Custo
- SB1: Descrição Genérica do Produto
- SB2: Saldos Físico e Financeiros
- SCP: Solicitações ao Armazém
- SCQ: Pré-Requisições
- SC1: Solicitações de Compra
- SC7: Pedido de Compra / Autorização de Entrega
- SZZ: Usuários. (Tabela criada para a aplicação)

A [Figura 13](#) mostra o modelo relacional entre as tabelas.

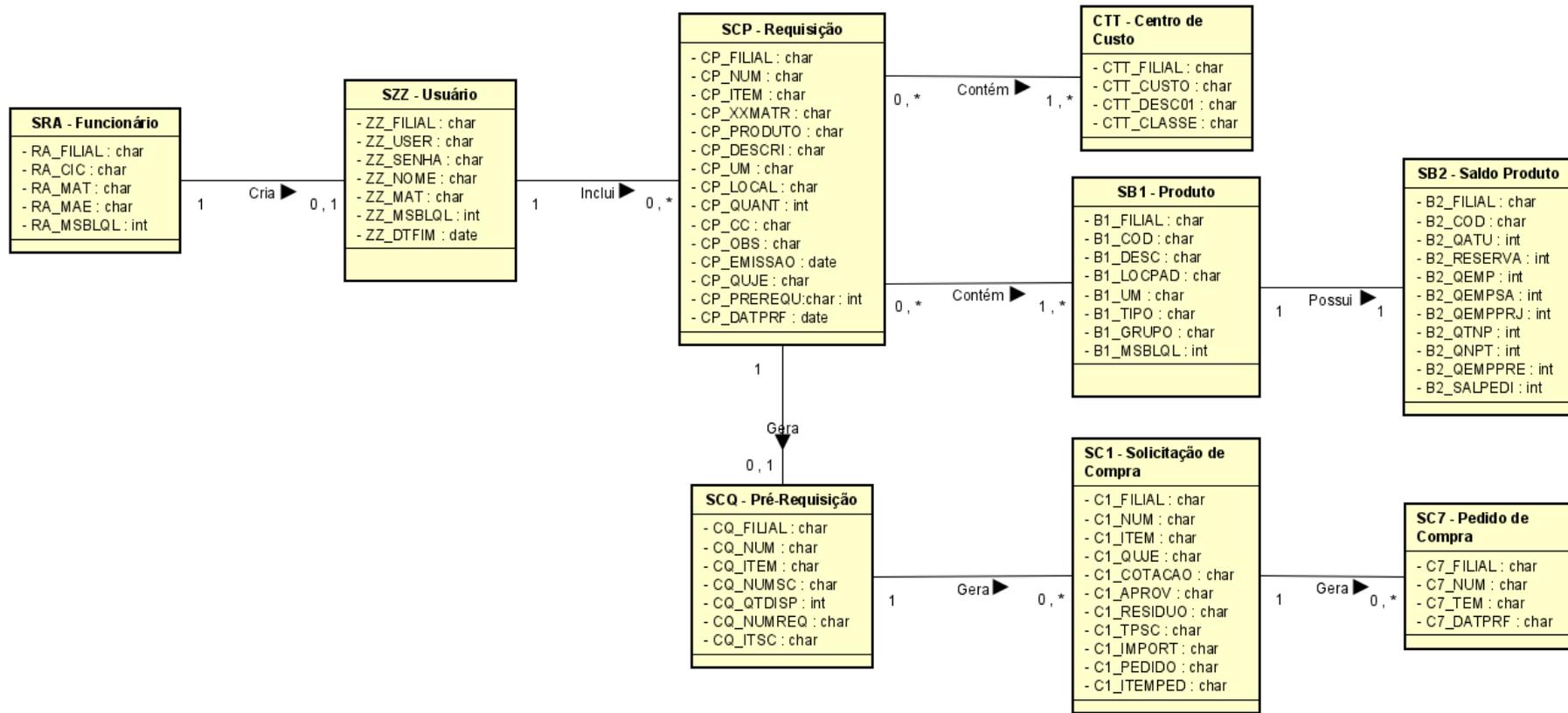


Figura 13 – Modelo Relacional do Banco de Dados.

Fonte: elaborado pela autora.

3.3.6 Arquitetura do Software

Uma característica marcante do *framework Laravel* é a arquitetura *Model-View-Controller* (MVC), que foi utilizada no nosso projeto e abre portas para complementações futuras no sistema. Os métodos dos *Controllers*, *Views* e as rotas foram amplamente utilizados, principalmente para a manipulação de dados das APIs e implementação do *Create, Read, Update, Delete* (CRUD) da solicitação e do *login*.

3.4 Implementação

O processo de implementação foi além do desenvolvimento de interface da aplicação *web*, também foi necessário o desenvolvimento das APIs no Protheus e alguns ajustes no Protheus.

3.4.1 Pré-Requisito

Para que o trabalho desenvolvido seja aderido por uma empresa é necessário o atendimento de alguns pré-requisitos, inicialmente a empresa precisa ter o ERP Protheus implementado na empresa e utilizar os módulos de Estoque/Custos e Compras.

Também é preciso que esteja disponível uma licença no Protheus que permita o uso da arquitetura *RESTful API*, e que o serviço de REST seja criado. Para criar o serviço REST no Protheus basta seguir o tutorial disponibilizado pela Totvs [neste link](#). Caso a empresa já possua o serviço REST criado e utilizado para outras aplicações este trabalho pode aproveitar do mesmo serviço. Além disso, é preciso que sejam realizadas algumas adequações no Protheus, conforme o [Apêndice B](#).

3.4.2 APIs

Para disponibilizar os dados do Protheus à aplicação web através do serviço REST, foram criados 5 *endpoints* de API, que serão responsáveis por receber e retornar às chamadas de API sempre que forem solicitadas pela aplicação web. A estruturação delas pode ser observada na [Figura 14](#). Em que cada API tem uma pasta, e dentro dela está o fonte principal da API e seus auxiliares.

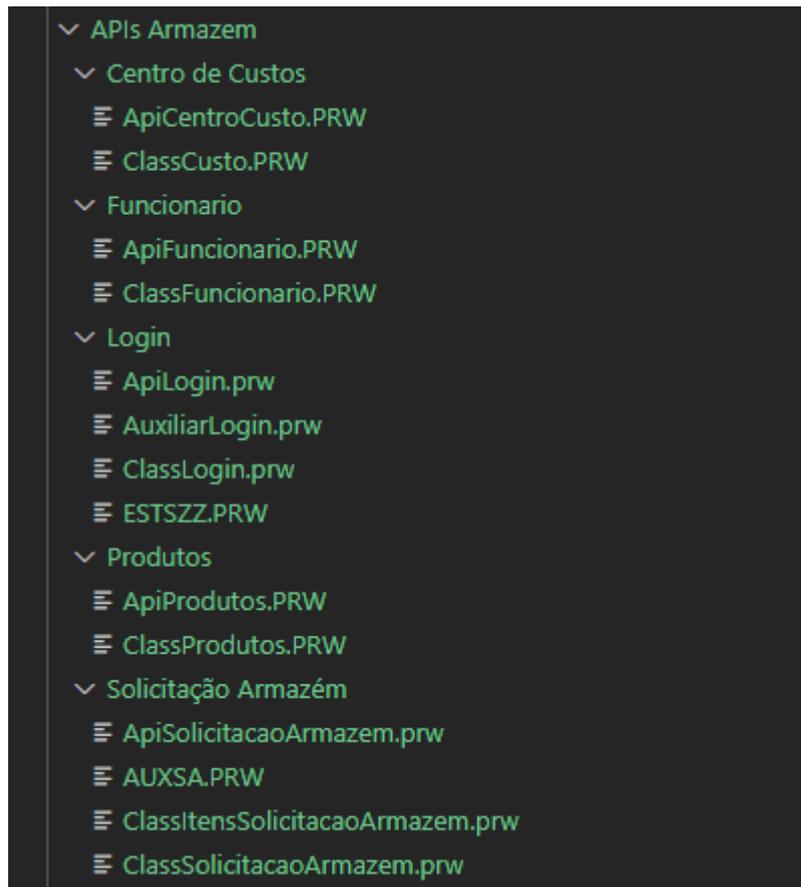


Figura 14 – Estruturação das APIS.

Fonte: elaborado pela autora.

- API de Centro de Custo: disponibiliza a consulta de todos os centros de custos cadastrados no sistema. Ela foi estruturada com o fonte `ApiCentroCusto` que faz a manipulação e a consulta principal e o fonte `ClassCusto` que possui uma classe que estrutura o retorno da API.
- API de Produtos: retorna a consulta de todos os produtos disponíveis, e permite o uso de diferentes tipos de filtros. Estruturada com o fonte `ApiProdutos` que faz a manipulação e a consulta principal, além da consulta com os filtros combinados, e o fonte `ClassProduto`, que possui a classe que estrutura o retorno da API.
- API de Funcionário: retorna a consulta de um usuário e suas verificações, confirmando se este existe ou não o funcionário no sistema. Também estruturada por um fonte que faz a manipulação e a consulta principal, o `ApiFuncionario`, e o fonte que possui a classe que estrutura o retorno da API, o fonte `ClassFuncionario`.
- API de *Login*: permite a verificação de usuário e senha, o cadastro de novo usuário e a alteração de senha de um usuário existente. Para isso, conta com o fonte `ApiLogin`, que faz a manipulação principal das funcionalidades GET, POST e PUT, e realiza a

consulta de verificação. O fonte AuxiliarLogin, que realiza a inserção ou alterações, e o fonte ClassLogin que possui a classe que estrutura o retorno do GET.

- API de Solicitação ao Armazém: permite a consulta de todas as solicitações de um usuário, a aplicação de filtros na consulta das solicitações, a inclusão, alteração ou exclusão de uma solicitação. Estruturada com o fonte ApiSolicitacaoArmazem, que faz a identificação da funcionalidade requisitada: GET, POST, PUT ou DELETE, e realiza a consulta de solicitações e seus filtros. O fonte AUXSA, que realiza a inserção, alteração ou exclusão da solicitação. E o fonte ClassSolicitacaoArmazem que contém a classe que estrutura o retorno do GET.

3.4.3 Interface Gráfica

A interface gráfica da aplicação *web* foi desenvolvida de modo a respeitar os fluxos apresentados e proporcionar uma navegação intuitiva e agradável ao usuário, de forma simplificada e direta.

O desenvolvimento foi feito utilizando as ferramentas mencionadas com o objetivo de aproveitar ao máximo as bibliotecas e funções padrões dos *frameworks* selecionados. Entretanto, algumas personalizações foram feitas através do *css* e *javascript*, com o objetivo de deixar o *layout* mais agradável e harmonioso.

3.4.4 Versionamento

O versionamento de todo o trabalho foi realizado com o *GitHub* que pode ser acessado [clikando aqui](#). E está apresentado conforme a [Figura 15](#), em que na pasta "armazem" está alocado os fontes da interface e na pasta Protheus estão as APIs desenvolvidas e o fonte de criação da rotina de usuários do armazém.

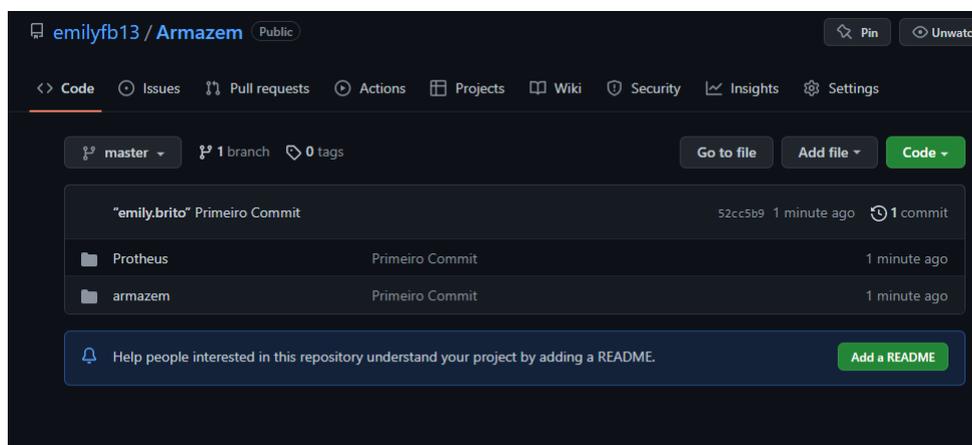


Figura 15 – Versionamento no *GitHub*.

Fonte: elaborado pela autora.

3.5 Testes

Durante o desenvolvimento foram realizados testes manuais tanto das APIs, quanto do funcionamento. Por se tratarem de testes simples não se viu necessidade de automação destes testes. E ao final do desenvolvimento e dos testes internos, foi realizado o teste de usabilidade com os funcionários do estudo de caso.

3.5.1 Testes das APIs

Por se tratarem de poucas APIs e com funcionalidades bem específicas, foi utilizado o *Postman* manualmente para os testes funcionais. Foram criadas consultas chaves para cada funcionalidade e eram usadas sempre que era acrescentado algo na API, necessitando novo teste.

No exemplo da [Figura 16](#) estão as consultas utilizadas na API de Solicitação ao Armazém, a API mais complexa desenvolvida, utilizando todas funções disponíveis: GET, POST, PUT, DELETE. As consultas das demais Apis seguem a estrutura parecida e podem ser encontradas no fonte da aplicação, na pasta `app/Http/Repositorios`.

3.5.2 Avaliação da Experiência do Usuário

Após a implementação final da aplicação, os dois funcionários consideradas chaves no estudo de caso foram novamente convidadas a realizar uma avaliação da experiência do usuário. Todos os dois são funcionários que realizam muitas requisições ao armazém, o primeiro funcionário realizada muitas requisições de materiais que costumam ter estoque no armazém, já o outro realiza mais requisições de produtos que necessitam de solicitação de compra.

Ambos os funcionários avaliaram positivamente o sistema e tiveram um resultado positivo no teste de usabilidade. Sem ajuda, conseguiram realizar as principais tarefas do sistema e encontraram a maioria das informações que desejavam.

O funcionário que realiza mais requisições de produtos em estoque destacou o quanto o histórico das suas requisições será importante. Atualmente ele realiza a anotação manual em planilhas de tudo que retira do armazém, para controle interno do seu departamento. Com o sistema ele poderá acessar todas as suas requisições e realizar diferentes tipos de filtragens.

Para o funcionário que requisita mais produtos técnicos e que normalmente necessitam de solicitação de compra, a grande vantagem do sistema é a listagem dos produtos. Por se tratar de produtos técnicos, ao solicitar no papel para o armazém, sem acesso a lista de produtos, muitas vezes as informações inseridas são confundidas pelos funcionários

```

-----GET-----
-> Filtro por ID:
http://XXX.XXX.XXX.XXX:XXXX/api/ApiSolicitacaoArmazem?ID=000001

-> Filtro por Solicitante:
http://XXX.XXX.XXX.XXX:XXXX/api/ApiSolicitacaoArmazem?SOLICITANTE=000002

-> Filtro por SORT e ORDER:
http://XXX.XXX.XXX.XXX:XXXX/api/ApiSolicitacaoArmazem?SOLICITANTE=000002&SORT=ID&ORDER=DESC

-> Filtros do Dashboard:
http://XXX.XXX.XXX.XXX:XXXX/api/ApiSolicitacaoArmazem?SOLICITANTE=000002&SORT=ID&ORDER=DESC&DASH=INCLUIDO
http://XXX.XXX.XXX.XXX:XXXX/api/ApiSolicitacaoArmazem?SOLICITANTE=000002&SORT=ID&ORDER=DESC&DASH=FIM
http://XXX.XXX.XXX.XXX:XXXX/api/ApiSolicitacaoArmazem?SOLICITANTE=000002&SORT=ID&ORDER=DESC&DASH=PC
http://XXX.XXX.XXX.XXX:XXXX/api/ApiSolicitacaoArmazem?SOLICITANTE=000002&SORT=ID&ORDER=DESC&DASH=DISP

-> Filtro Avançado:
http://XXX.XXX.XXX.XXX:XXXX/api/ApiSolicitacaoArmazem?SOLICITANTE=000002&SORT=ID&ORDER=DESC&ID=000001
&DASH=PC&CODPROD=MC001&PROD=CANETA&OBS=teste&EMISSAOINI=20221027&EMISSAOFIM=20221027

-----POST-----
Json:

{ "ITENS": [ { "SOLICITANTE": "000002", "CENTROCUSTO": "10225", "PRODUTO": "MC005", "QUANTIDADE": 1, "OBSERVACAO": "" },
  { "SOLICITANTE": "000002", "CENTROCUSTO": "10309", "PRODUTO": "MC001", "QUANTIDADE": 1, "OBSERVACAO": "01" },
  { "SOLICITANTE": "000002", "CENTROCUSTO": "10225", "PRODUTO": "MC002", "QUANTIDADE": 3, "OBSERVACAO": "" } ] }

Retorno:
{
  "ID": "000001",
  "SUCESSO": "1",
  "OBS": "SUCESSO!"
}

-----PUT-----
Json:

{"ID": "000001", "ITENS":
  [ { "SOLICITANTE": "000002", "ITEM": "02", "CENTROCUSTO": "10225", "PRODUTO": "MC004", "QUANTIDADE": 2, "OBSERVACAO": "teste" },
    { "SOLICITANTE": "000002", "ITEM": "03", "CENTROCUSTO": "10225", "PRODUTO": "MC003", "QUANTIDADE": 5, "OBSERVACAO": "teste 2" }
  ] }

Retorno:
{
  "ID": "000001",
  "SUCESSO": "1",
  "OBS": "SUCESSO!"
}

-----DELETE-----
Json:
{ "ID": "000001", "ITENS": [ {"ITEM": "02"}, {"ITEM": "01"} ] }

Retorno:
{
  "ID": "000001",
  "SUCESSO": "1",
  "OBS": "SUCESSO!"
}

```

Figura 16 – Exemplos das consultas no Postman.

Fonte: elaborado pela autora.

do armazém, causando a compra de produtos diferentes do solicitado. Com a lista ele consegue acessar a descrição detalhada do produto e pedir exatamente o que precisa.

4 Resultados

Este capítulo apresenta os resultados obtidos através do desenvolvimento deste trabalho e mostra a aplicação web desenvolvida. A [Figura 17](#) mostra o fluxo final da aplicação web, por onde o solicitante realizará sua requisição, juntamente ao fluxo de atendimento no Protheus, no qual o Atendente realizará o atendimento e entrega da requisição.

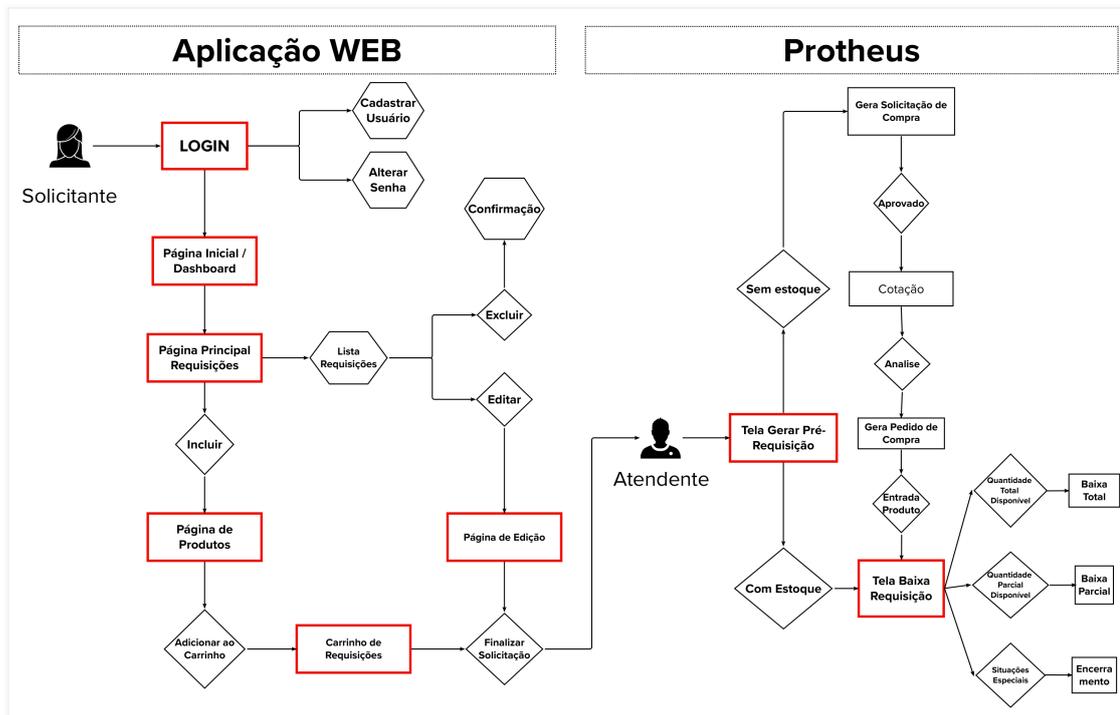


Figura 17 – Fluxo geral do processo implementado.

Fonte: os autores.

4.1 Login

Ao acessar a página da aplicação, a primeira tela apresentada é a de *login* conforme a [Figura 18](#). Nela é possível preencher os dados para acesso: com o usuário (CPF) e a senha, caso o usuário já tenha sido cadastrado. Na mesma tela é possível ainda, clicar no botão "Cadastre-se" e no botão "Alterar Senha".

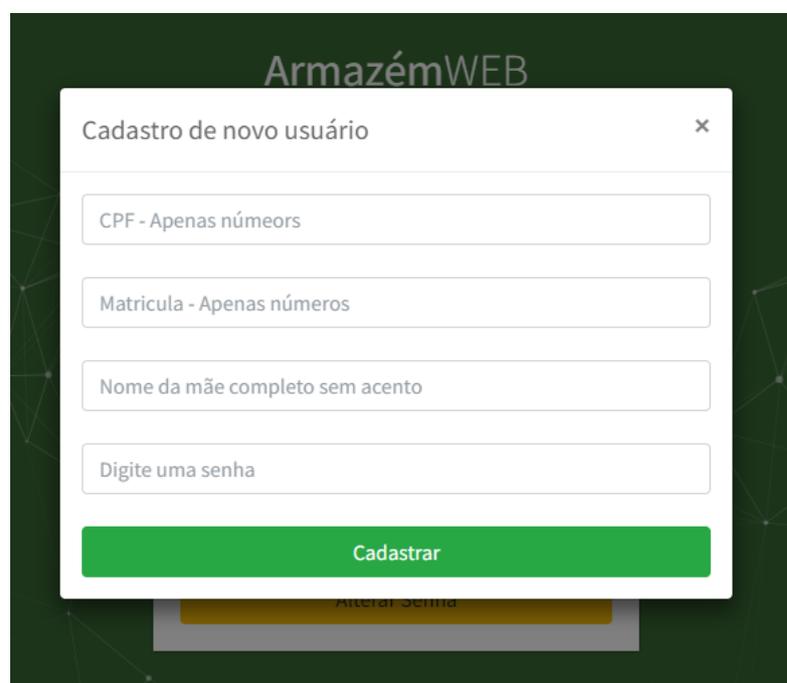
Ao clicar no botão "Cadastre-se" será aberto o box de perguntas mostrado na [Figura 19](#), nele será solicitado o preenchimento dos campos: CPF, apenas com os números, matrícula, também apenas os números, nome da mãe e uma senha. Ao preencher esses campos e clicar em Cadastrar, será feita uma verificação se todos os campos foram preenchidos, caso algum não tenha sido preenchido será solicitado o preenchimento.



A screenshot of the login page for 'ArmazémWEB'. The page has a dark green background with a white geometric pattern. At the top, the title 'ArmazémWEB' is displayed in white. Below the title, a white box contains the text 'Informe suas credencias para acessar o sistema'. There are three input fields: 'Nome de Usuário (CPF)' with a person icon, 'Senha' with a lock icon, and a green 'Entrar' button. Below these are two more buttons: a blue 'Cadastre-se' button and a yellow 'Alterar Senha' button.

Figura 18 – Tela de *login* da aplicação.

Fonte: os autores.



A screenshot of the registration page for 'ArmazémWEB'. The page has a dark green background with a white geometric pattern. At the top, the title 'ArmazémWEB' is displayed in white. Below the title, a white box contains the text 'Cadastro de novo usuário' and a close button (X). There are four input fields: 'CPF - Apenas númeors', 'Matricula - Apenas números', 'Nome da mãe completo sem acento', and 'Digite uma senha'. Below these is a green 'Cadastrar' button.

Figura 19 – Tela de cadastro da aplicação.

Fonte: os autores.

Caso todos os campos tenham sido preenchidos, o sistema fará uma verificação se o CPF e senha preenchido já estão cadastrados, se o usuário e senha não forem encontrados o sistema seguirá para a verificação do funcionário. Em que será analisado se a combinação

de CPF, matrícula e nome da mãe existe na tabela de funcionários e se este está ativo.

Na hipótese de retorno positivo o cadastro de usuário com o CPF e senha digitados será realizado e o usuário poderá usar esses dados para realizar o *login* no sistema, caso contrário, retornará uma mensagem de erro.

Ao clicar no botão "Alterar senha" será exibido o box de perguntas apresentado na Figura 20, onde será solicitado o preenchimento dos campos similar ao box de cadastro, com exceção de que agora será digitada uma nova senha.

Neste caso, o sistema realizará também a verificação da existência do funcionário com a combinação de campos e em seguida verificará a existência do CPF na tabela de usuários, caso encontre o CPF, será realizada a alteração da senha e retornará uma mensagem de sucesso ao usuário, caso contrário será exibida uma mensagem de erro.

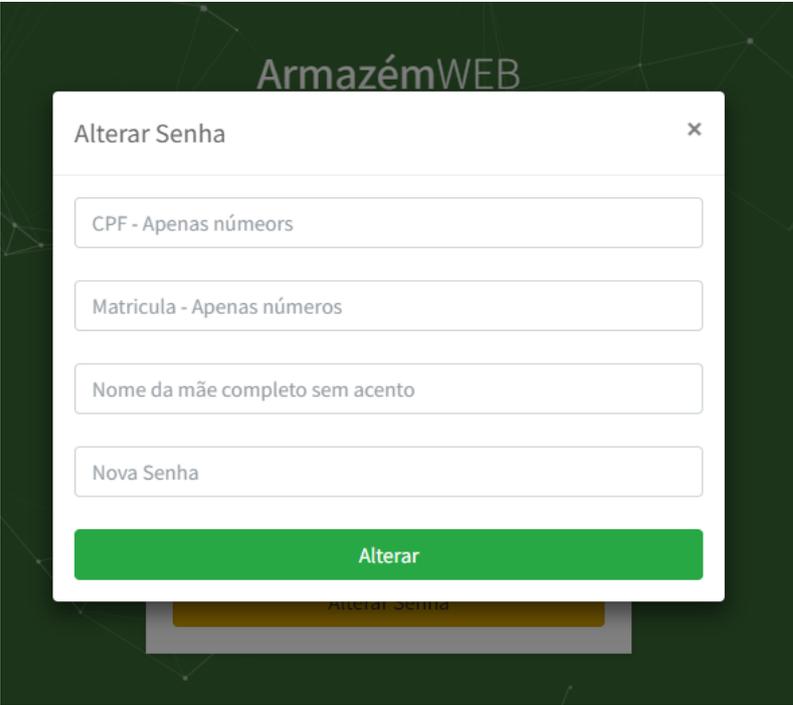
A imagem mostra uma interface de usuário para a aplicação "ArmazémWEB". No topo, o nome "ArmazémWEB" é exibido em uma fonte branca sobre um fundo verde escuro. Abaixo, há um formulário modal branco com o título "Alterar Senha" e um ícone de fechar (X) no canto superior direito. O formulário contém quatro campos de entrada de texto: "CPF - Apenas númeors", "Matricula - Apenas números", "Nome da mãe completo sem acento" e "Nova Senha". Abaixo dos campos, há um botão verde com o texto "Alterar".

Figura 20 – Tela de alteração de senha da aplicação.

Fonte: os autores.

4.2 Página Inicial

Ao realizar o *login*, o usuário será direcionado a página inicial, apresentada na Figura 21. Nesta página é exibido o menu lateral "Requisição ao Armazém" que possui os submenus "Requisições", que leva até a página das requisições e o submenu "Incluir Nova Requisição", que leva diretamente a página de seleção de produtos.

A barra superior possui um ícone para encolher esse menu lateral, o ícone de usuário que ao clicar apresenta o botão de sair, e o último ícone que é para expandir a tela.

No conteúdo apresenta um *dashboard* que faz o resumo de todas as suas requisições, dividindo-as em 4 status. Cada box é resultado de uma consulta a API "SolicitacaoArmazem" com o filtro de usuário e status aplicado, mostrando a quantidade de requisições que possuem itens com aquele status. Ao clicar em "Mais informações" você é encaminhado para a página de requisições com o filtro aplicado para mostrar apenas as requisições que possuem itens no status que você selecionou.

Mais a baixo possui um box para inserir informações de contato com a equipe do suporte do Protheus e também a opção de inserir um formulário anônimo.

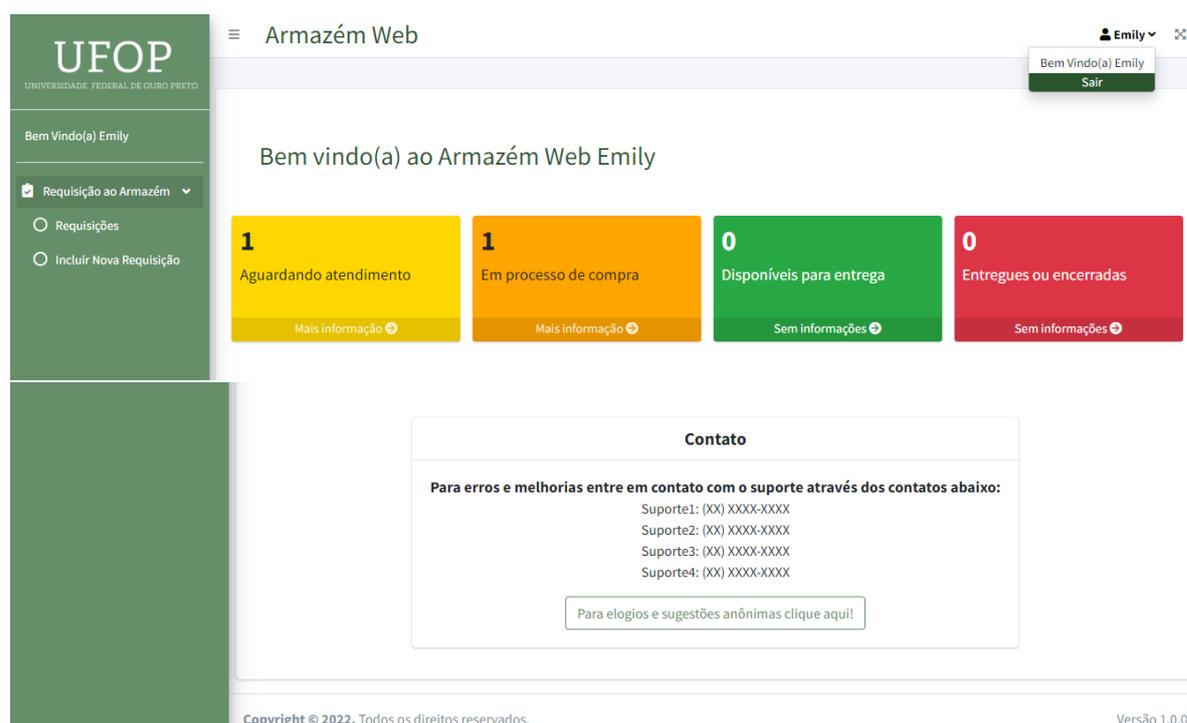


Figura 21 – Tela inicial da aplicação.

Fonte: os autores.

4.3 Página de Requisições

A página de requisições pode ser acessada através do submenu "Requisições" ou através do botão "Mais informações" dos *dashboard*. Na [Figura 22](#) é apresentada a página sem filtros de status, ela é resultado de uma consulta a API "ApiSolicitacaoArmazem" com filtro aplicado para que seja exibida apenas as requisições do usuário logado.



Figura 22 – Página de Requisições.

Fonte: os autores.

Na barra superior passa a ser exibido além do ícone de sair e expandir, também o ícone de carrinho, que ao clicar você acessa a página Carrinho de Requisições, e o ícone de informação, que dá acesso a uma página de tutorial de uso das requisições ao armazém.

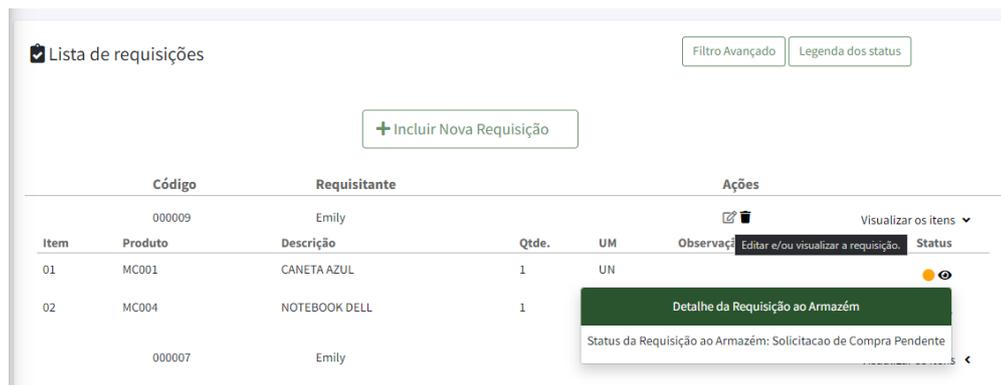
Logo abaixo, é possível observar o caminho entre as páginas, no caso *Home / Requisição ao Armazém*, em que *Requisição ao Armazém* é a página atual e *Home* é um *link* clicável que leva até a página principal. Durante toda a navegação entre as páginas um caminho como este está disponível para o usuário poder voltar as páginas anteriores com facilidade.

No conteúdo encontra-se no canto direito os botões "Filtro Avançado" e "Legenda dos status", e abaixo de forma central o botão "Incluir Nova Requisição". Em seguida, possui a listagem com todas as requisições já feitas pelo usuário, sendo possível visualizar o número da requisição, o nome do requisitante, as ações (editar e excluir) e a opção de visualizar os itens. Por fim, está a paginação para situações em que a lista de requisições ultrapasse o espaço de uma página.

Ao clicar em "Visualizar os Itens" de uma requisição é listado logo abaixo todos os itens daquela solicitação e as suas principais informações, conforme a [Figura 23](#). Para visualizar todos dados completos, o usuário pode clicar no ícone de editar a requisição que a página exibirá todos os dados completos.

Ao final da linha do item está os ícones de status, no qual o usuário pode acompanhar o status em que está o processo de requisição e ao clicar nos ícones pode visualizar o detalhamento daquele status. A legenda de cada cor do status pode ser encontrada ao clicar no botão "Legenda dos status", que irá exibir uma janela conforma a [Figura 24](#).

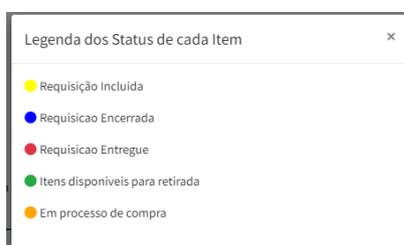
O detalhamento da Requisição ao Armazém passa por alguns estágios, de acordo



Código	Requisitante	Ações				
000009	Emily	Visualizar os itens				
Item	Produto	Descrição	Qtde.	UM	Observação	Status
01	MC001	CANETA AZUL	1	UN		
02	MC004	NOTEBOOK DELL	1			
000007	Emily					

Figura 23 – Itens das requisições.

Fonte: os autores.



Legenda dos Status de cada Item
● Requisição Incluída
● Requisicao Encerrada
● Requisicao Entregue
● Itens disponiveis para retirada
● Em processo de compra

Figura 24 – Janela de Legenda dos status de requisições.

Fonte: os autores.

com o processo. No fluxo da [Figura 25](#) pode-se observar todos os possíveis estágios de detalhamento que o processo pode ter e sua respectiva cor.

Para realizar uma filtragem dessas requisições, o usuário pode acessar através do *dashboard*, ou então clicar no botão "Filtro Avançado" que exibirá uma janela conforme a [Figura 26](#). Nela, o usuário pode escolher os filtros que desejar, e realizar combinações entre eles.

Ao aplicar o filtro, o sistema realiza uma consulta pela API "ApiSolicitacaoArmazem" com todos os filtros existentes nela e os conteúdos selecionados pelo usuário, retornando as requisições já filtradas, além disso a aplicação passa a exibir o botão "Limpar filtro", conforme a [Figura 27](#), que permite o usuário a cancelar os filtros aplicados e visualizar todas as suas requisições novamente.

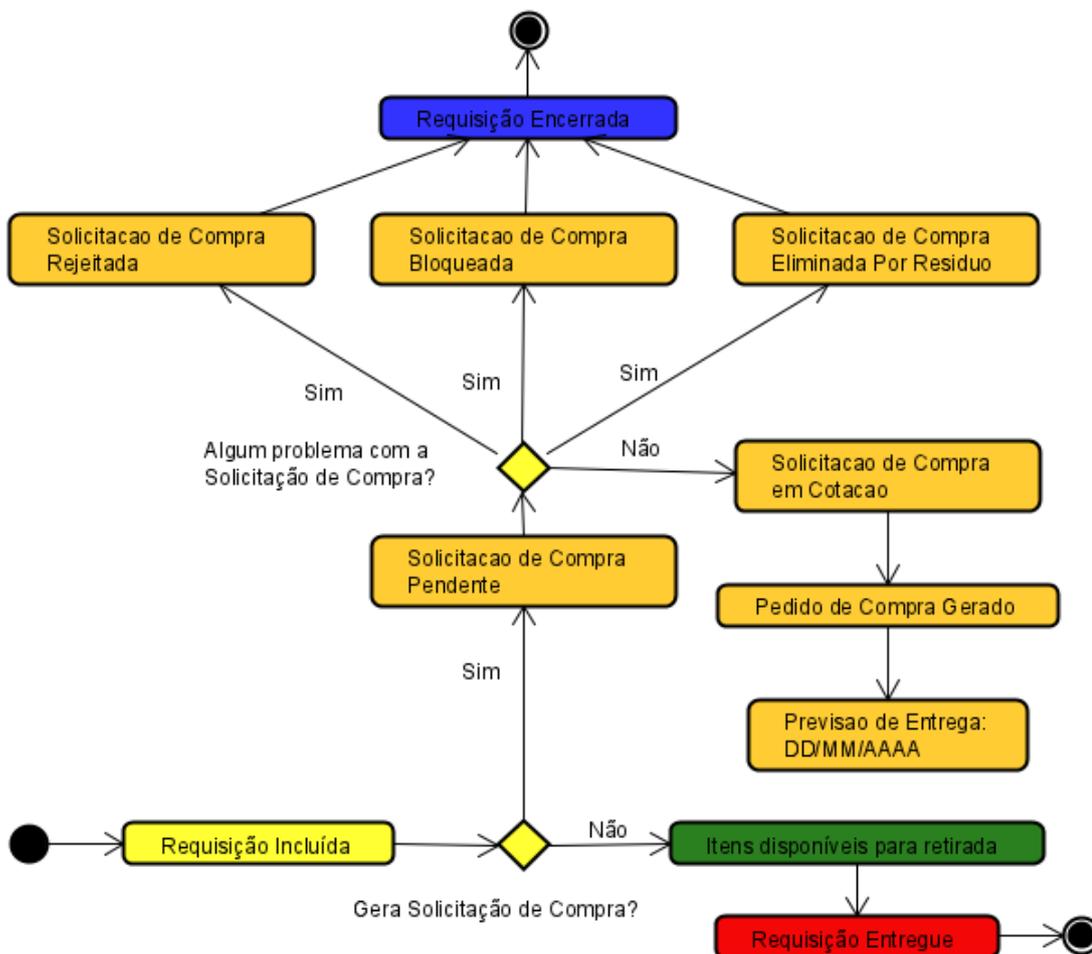


Figura 25 – Fluxo de status da requisição.

Fonte: os autores.

Filtrar Requisições ×

Código da Solicitação

Status

Data Emissão

Código do Produto

Produto

Observação

*Nenhum campo é de preenchimento obrigatório.

Figura 26 – Janela de filtros da Página de Requisições.

Fonte: os autores.



Figura 27 – Página de Requisições com filtro aplicado.

Fonte: os autores.

4.4 Inclusão de uma Nova Requisição

Para incluir uma nova requisição você pode clicar no submenu lateral "Incluir Nova Requisição" ou no botão "Incluir Nova Requisição", conforme mostra a [Figura 28](#), nela também vale ressaltar a mensagem apresentada caso o usuário ainda não tenha feito nenhuma requisição.

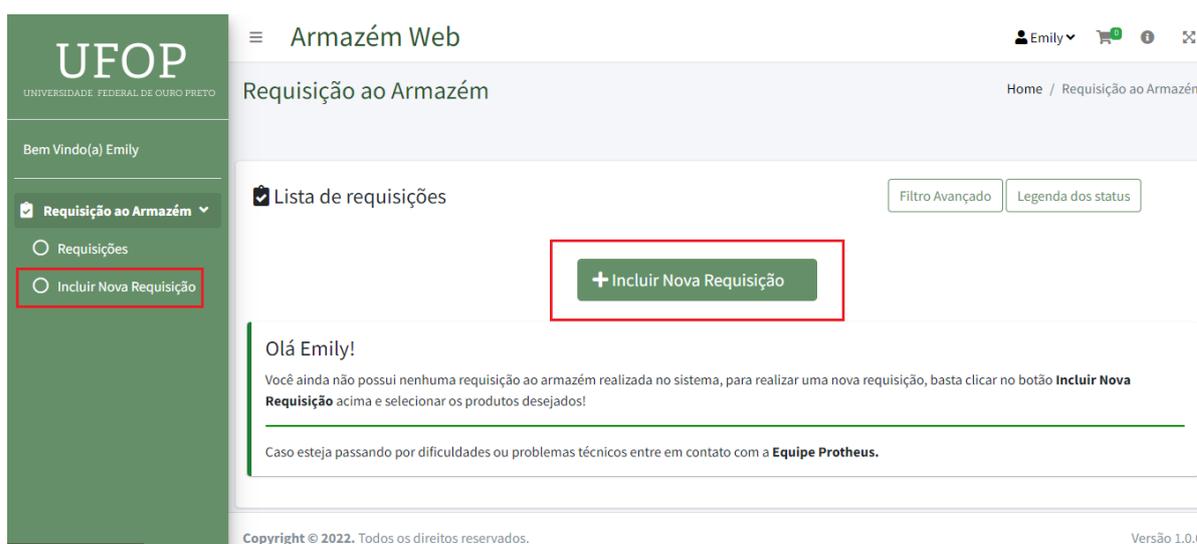


Figura 28 – Página de Requisições.

Fonte: os autores.

Ao clicar, o usuário será encaminhado para a página de Produtos, exibida na [Figura 29](#), na qual ele poderá adicionar ao carrinho os produtos que ele desejar.

A página é resultado da consulta a API "ApiProdutos", sem filtros e exibirá todos os produtos cadastros e ativos do sistema. Na listagem é possível verificar o código do produto, sua descrição, o saldo, tipo e grupo. O saldo disponível é calculado de acordo com o saldo em estoque subtraindo todos os possíveis empenhos que ele possa ter.

Para adicionar ao carrinho basta clicar no ícone de carrinho de compras no final da lista do produto e à medida que os produtos são adicionados a aplicação faz a gravação

The screenshot shows the 'Armazém Web' interface. The header includes the UFOP logo and the text 'UNIVERSIDADE FEDERAL DE OURO PRETO'. The main title is 'Armazém Web' and the page title is 'Requisição ao Armazém'. The user is logged in as 'Emily'. The page content is titled 'Lista de produtos' and features a search bar with 'Palavra-chave' and a 'Filtro Avançado' button. Below the search bar is a table with the following data:

Código	Descrição	Saldo	Tipo	Grupo
AI001	SACHEZEIRA BOMBA TUDO	2	AI	
MANUTENCAO	ITEM PARA CONTROLE DE MANUTENC	0	MC	
MC001	CANETA AZUL	100	MC	0008
MC002	CANETA PRETA	100	MC	0008
MC003	CANETA VERMELHA	0	MC	
MC004	NOTEBOOK DELL	0	MC	
MC005	FONE DE OUVIDO	0	MC	

At the bottom of the table is a button labeled 'Ir Para o Carrinho'. A tooltip 'Adicionar ao Carrinho' is visible over the cart icon for the first product. The footer contains 'Copyright © 2022. Todos os direitos reservados.' and 'Versão 1.0.0'.

Figura 29 – Página de Produtos.

Fonte: os autores.

destes produtos na sessão e exibe a quantidade no ícone de carrinho da barra superior.

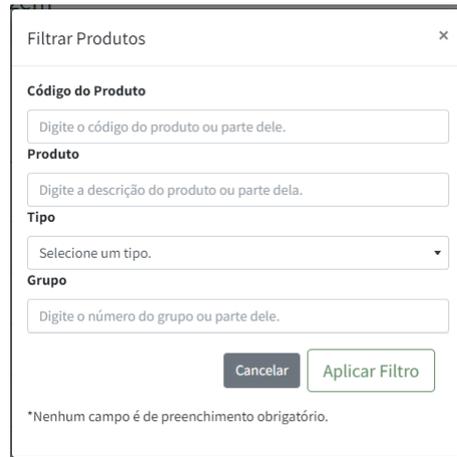
Ao final da página está a paginação para casos em que o número de produtos ultrapasse a quantidade permitida em uma página. Na parte superior do conteúdo está o campo de busca, por onde pode-se pesquisar os produtos por qualquer palavra-chave seja do código, descrição, grupo ou tipo, e a aplicação retornará a listagem filtrada e o botão "Limpar Filtro".

Ao lado está o botão "Filtro Avançado", que ao clicar exibirá a tela conforme [Figura 30](#), nela, assim como a tela de filtro de requisições, o usuário poderá fazer diferentes tipos de filtragens combinadas. Essas filtragens serão feitas via a API "ApiProdutos" e a aplicação também exibirá o botão "Limpar Filtro".

Finalizando as adições de produtos no carrinho o usuário poderá acessar a página Carrinho de Requisições utilizando o ícone de carrinho da barra superior ou o botão "Ir Para o Carrinho" que está localizado no final da página.

Feito isso, será apresentada a página Carrinho de Requisições, conforme a [Figura 31](#), com todos os itens adicionados na página anterior. Nesta página o usuário deverá informar a quantidade e centro de custo, e poderá colocar uma observação, ou então, remover aquele produto do carrinho.

Para selecionar um centro de custo é necessário selecioná-los dentre as opções que são exibidas através da busca pela API "ApiCentroCustos", no seletor também é possível fazer uma busca pelo código ou nome do centro de custo, como mostra a [Figura 32](#)



Filtrar Produtos

Código do Produto

Produto

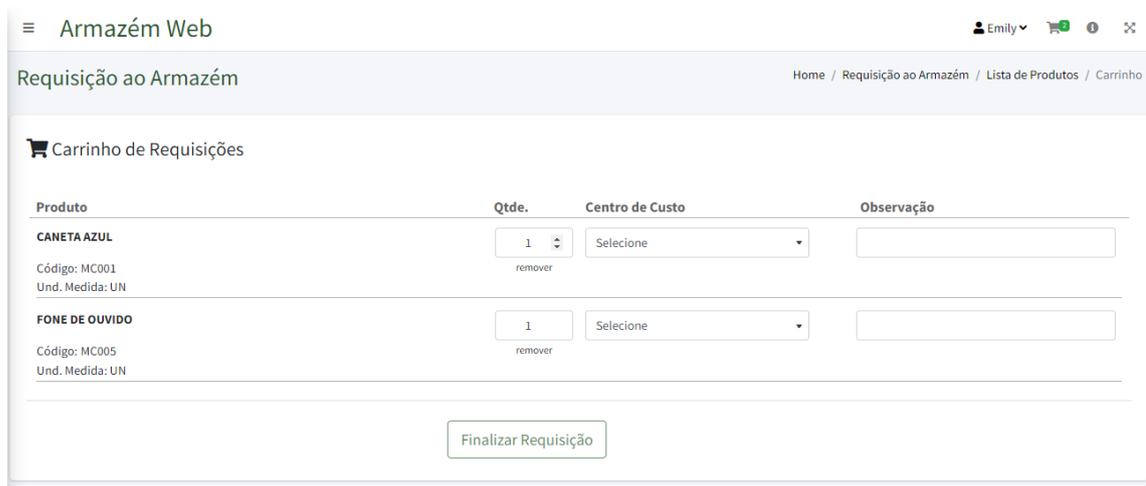
Tipo

Grupo

*Nenhum campo é de preenchimento obrigatório.

Figura 30 – Janela de filtros da Página de Produtos.

Fonte: os autores.



Armazém Web

Emily

Requisição ao Armazém

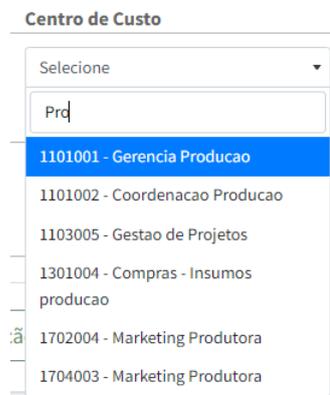
Home / Requisição ao Armazém / Lista de Produtos / Carrinho

Carrinho de Requisições

Produto	Qtde.	Centro de Custo	Observação
CANETA AZUL Código: MC001 Und. Medida: UN	1 remover	Selecione	<input type="text"/>
FONE DE OUVIDO Código: MC005 Und. Medida: UN	1 remover	Selecione	<input type="text"/>

Figura 31 – Página do Carrinho de Requisições.

Fonte: os autores.



Centro de Custo

Selecione

Prq

- 1101001 - Gerencia Producao
- 1101002 - Coordenacao Producao
- 1103005 - Gestao de Projetos
- 1301004 - Compras - Insumos producao
- 1702004 - Marketing Produtora
- 1704003 - Marketing Produtora

Figura 32 – Seletor de Centro de Custos.

Fonte: os autores.

Ao finalizar os preenchimentos necessários basta o usuário clicar no botão "Finalizar

Requisição" que a nova requisição será cadastrada via a API "ApiSolicitacaoArmazem" e a aplicação retornará para a página inicial onde a nova requisição já estará disponível para visualização.

4.5 Editar e excluir uma requisição

Só é possível editar ou excluir uma requisição que ainda está no primeiro status, Requisição Incluída, pois ainda não foi gerada a pré-requisição, depois que ela é gerada o processo deve ser cancelado pelo armazém diretamente no Protheus, da forma correta de acordo com cada processo.

Para editar uma requisição, basta clicar no ícone de lápis que está na coluna Ações de cada requisição e o usuário será transferido para a página de edição, que vem auto incrementada com todos os seus valores, como na [Figura 33](#). Nesta página o usuário poderá editar os campos e/ou remover itens. Caso a requisição já tenha gerado pré-requisito, a página será aberta com os campos bloqueados para edição como na [Figura 34](#).

Produto	Qtde.	Centro de Custo	Necessi.	Observação
CANETA 07 ESCRITA FINA PRETA - Código: MC150600003 Und. Medida: UN Data de Emissão da Solicitação: 14/10/22	2 remove	10328 - PROTHEUS FABRICA	Noi	Teste
NOTEBOOK DELL LATITUDE 3540 Código: AI060500183 Und. Medida: UN Data de Emissão da Solicitação: 14/10/22	1 remove	10328 - PROTHEUS FABRICA	Noi	Teste

Finalizar Alteração

Figura 33 – Página de edição.

Fonte: os autores.

Requisição ao Armazém Home / Requisição ao Armazém / Edição

Atenção!
Olá **Emily**, essa solicitação não pode mais ser alterada pois não está com o status de **Requisição Incluída**, você está apenas no modo de visualização!

[✎](#) Editar Requisição

Produto	Qtde.	Centro de Custo	Necessi.	Observação
CANETA 07 ESCRITA FINA PRETA - Código: MC150600003 Und. Medida: UN Data de Emissão da Solicitação: 14/10/22	2	10328 - PROTHEUS FABRICA	Normi	Teste
NOTEBOOK DELL LATITUDE 3540 Código: AI060500183 Und. Medida: UN Data de Emissão da Solicitação: 14/10/22	1	10328 - PROTHEUS FABRICA	Normi	Teste

[Finalizar Alteração](#)

Figura 34 – Página de edição apenas no modo visualização.

Fonte: os autores.

Para excluir, é necessário clicar no ícone de lixeira que está localizado na coluna Ações de cada requisição e será exibida a mensagem de confirmação como na [Figura 35](#). Caso a requisição não possa ser excluída por já ter gerado pré-requisição será exibida a mensagem da [Figura 36](#).

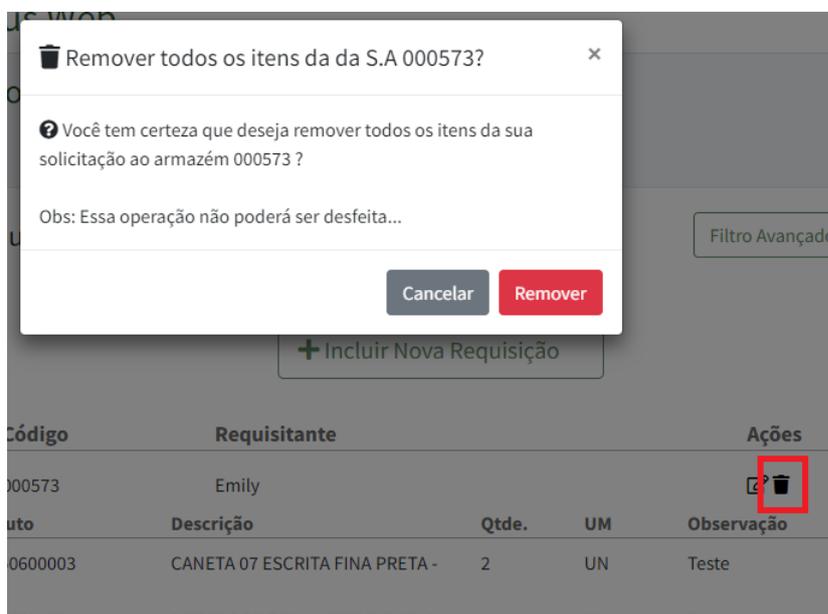


Figura 35 – Janela de confirmação para excluir requisição.

Fonte: os autores.

⚠️ Atenção! Só é possível deletar uma S.A com o status de: Requisição Incluída!

Lista de requisições Filtro Avançado Legenda dos status

[+ Incluir Nova Requisição](#)

Código		Requisitante	Ações			
Item	Produto	Descrição	Qtde.	UM	Observação	Status
01	MC150600003	CANETA 07 ESCRITA FINA PRETA -	2	UN	Teste	●
02	AI060500183	NOTEBOOK DELL LATITUDE 3540	1	UN	Teste	●

Figura 36 – Erro ao excluir requisição.

Fonte: os autores.

5 Conclusão

Neste trabalho foi desenvolvida uma aplicação web para a realização de requisições ao armazém voltado para empresas que utilizam o ERP Protheus e buscam automatizar a inserção de informações do seu processo de forma econômica. A aplicação foi desenvolvida utilizando os *frameworks* *Laravel* e *Bootstrap*, tendo como modelo de painel de administração o *AdminLTE*.

O trabalho foi baseado no conceito padrão da solução existente no Protheus, juntamente com os pontos levantados durante o estudo de caso realizado. E utilizou da fundamentação de desenvolvimento web juntamente com os conceitos de desenvolvimento do Protheus.

Os trabalhos relacionados ilustraram a importância do uso de ERP em uma empresa, da gerência adequada dos estoques e as vantagens de se contratar um ERP que possua todas as ferramentas necessárias para a empresa centralizado em um só software. E por fim, trouxeram um ponto que também pode afetar a aderência deste trabalho nas empresas, que é a resistência cultural de uma empresa.

O desenvolvimento foi direcionado para a economia de licenças do Protheus e aproveitamento do Banco de Dados dele, de forma a respeitar o conceito de integração dos ERP. Além disso, a aplicação foi modelada de forma a possuir uma interface amigável ao usuário, trazendo conceitos similares ao de compras *online* e buscando ser o mais intuitivo possível.

Por se tratar de um projeto *open-source*, o trabalho possui algumas limitações. A principal é a implementação de meios de segurança, como por exemplo os métodos de criptografias e autenticação das APIs, já que são conceitos que cada empresa deve adotar de forma sigilosa. Além disso, não é possível prever as particularidades do processo de requisição ao armazém de cada empresa. Portanto não é possível criar limitações de acesso dos usuários a determinados tipos de produtos, nem implementar a rotina de liberação de requisição disponibilizada pela Totvs como opcional na solução de requisição ao armazém.

Em conclusão, o trabalho ilustrou a vantagem de contratar um ERP mais robusto e completo, a importância do gerenciamento dos estoques de uma empresa, mesmo que seja pequena, automatizar os processos da empresa e aproximá-los ao padrão de mercado. Além disso, comprovou-se que é possível aderir ao processo de requisição ao armazém do Protheus, de forma econômica e que alcance um público diversificado de funcionários.

5.1 Trabalhos Futuros

O trabalho foi criado de forma a oferecer uma plataforma escalável, que proporciona uma estrutura para suprir as limitações que o projeto open source teve, além de possibilitar melhorias e crescimento. Como ponto de melhoria é aconselhado as empresas que aderirem a este trabalho, incluir um cabeçalho de autenticação nas APIs, melhorar o método de criptografia utilizado nas senhas de usuários e limitar o acesso ao sistema apenas em redes internas. Além disso, pode-se aderir a rotina de liberação de requisições e utilizar dos pontos de entrada do Protheus para incluir as regras de negócio da empresa. Por fim, a estrutura criada também pode ser usada para a implementação de outros processos, inclusive de outros módulos, já que a base de dados é a mesma utilizada no Protheus.

Referências

- ADMINLTE. *AdminLTE Bootstrap Admin Dashboard Template*. 2022. Disponível em: <<https://adminlte.io/>>. Acesso em: 20 Out. 2022. Citado na página 25.
- ASTAH. *The power of software modeling*. 2022. Disponível em: <<https://astah.net/>>. Acesso em: 20 Out. 2022. Citado na página 26.
- BOOTSTRAP. *Build fast, responsive sites with Bootstrap*. 2022. Disponível em: <<https://getbootstrap.com/>>. Acesso em: 20 Out. 2022. Citado na página 25.
- FIELDING, R. *Architectural Styles and the Design of Network-based Software Architectures*. Tese (Doutorado) — University of California, 2000. Citado na página 19.
- GIT. *GitHub*. 2022. Disponível em: <<https://github.com/>>. Acesso em: 20 Out. 2022. Citado na página 25.
- JQUERY. *What is jQuery?* 2022. Disponível em: <<https://jquery.com/>>. Acesso em: 20 Out. 2022. Citado na página 25.
- LARAVEL. *Installation - Meet Laravel*. 2022. Disponível em: <<https://laravel.com/docs/9.x>>. Acesso em: 20 Out. 2022. Citado na página 24.
- MARTELLI, L. L.; DANDARO, F. Planejamento e controle de estoque nas organizações. *Revista Gestão Industrial*, Universidade Tecnológica Federal do Paraná - UTFPR, v. 11, n. 2, p. 170–185, 2015. Citado 2 vezes nas páginas 15 e 21.
- MICROSOFT. *SQL Server 2022*. 2022. Disponível em: <<https://www.microsoft.com/pt-br/sql-server>>. Acesso em: 20 Out. 2022. Citado na página 25.
- MURAL. *Let's transform teamwork*. 2022. Disponível em: <<https://www.mural.co/>>. Acesso em: 20 Out. 2022. Citado na página 26.
- PADULA, M. I. J. et al. *A importância do uso de software ERP como ferramenta para um eficiente controle de estoque*. São Paulo, 2021. Citado na página 20.
- PAVIN, R. D. P.; KLEIN, A. Z. *Organization consequences of the adoption of mobile ERP systems: Case studies in Brazil*. São Paulo, 2015. v. 3, 219-232 p. Citado na página 14.
- POSTMAN. *What is Postman?* 2022. Disponível em: <<https://www.postman.com/>>. Acesso em: 20 Out. 2022. Citado na página 26.
- SILVA, E. C. da; NAJM, L. Erp - módulo estoque/custos, pcp e os problemas burocráticos. *Revista Network Technologies*, Faculdades Network – Revista da Faculdade de Sistema de Informação, v. 7, n. 1, p. 54–65, 2014. Citado na página 22.
- TOTVS. *A TOTVS é líder absoluta em software de gestão*. 2022. Disponível em: <<https://www.totvs.com/sistema-de-gestao/totvs-backoffice-linha-protheus/>>. Acesso em: 20 Out. 2022. Citado na página 14.

TOTVS. *Tudo sobre o TOTVS BACKOFFICE – LINHA PROTHEUS*. 2022. Disponível em: <<https://produtos.totvs.com/ficha-tecnica/tudo-sobre-o-totvs-backoffice-linha-protheus/>>. Acesso em: 20 Out. 2022. Citado na página 18.

VISUAL STUDIO CODE. *Getting Started*. 2022. Disponível em: <<https://code.visualstudio.com/docs>>. Acesso em: 20 Out. 2022. Citado na página 26.

Apêndices

APÊNDICE A – Diagramas Complementares

A.1 Diagrama de Atividade

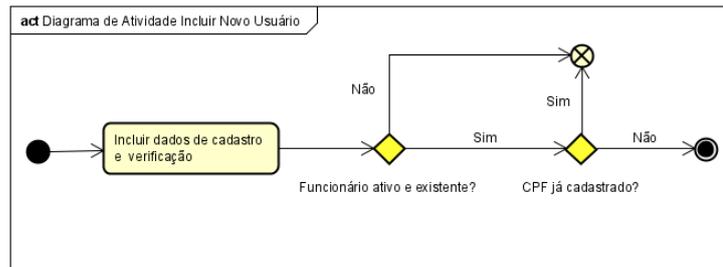


Figura 37 – Diagrama de Atividade de Cadastro de Usuário.

Fonte: elaborado pela autora.

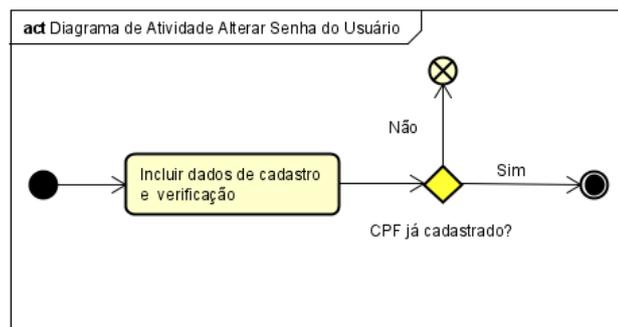


Figura 38 – Diagrama de Atividade de Alterar Senha de Usuário.

Fonte: elaborado pela autora.

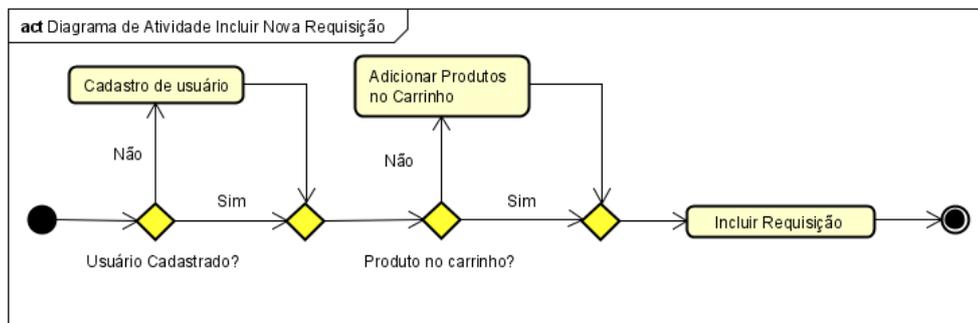


Figura 39 – Diagrama de Atividade de Incluir Nova Requisição.

Fonte: elaborado pela autora.

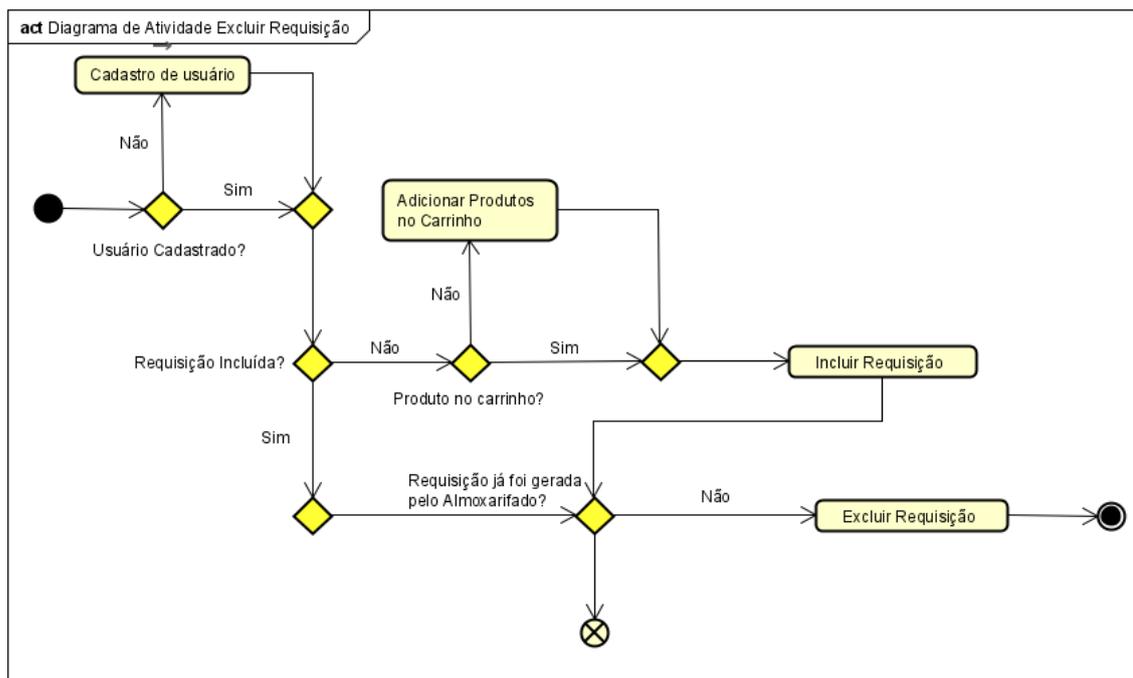


Figura 40 – Diagrama de Atividade de Excluir Requisição.

Fonte: elaborado pela autora.

A.2 Diagrama de Sequência

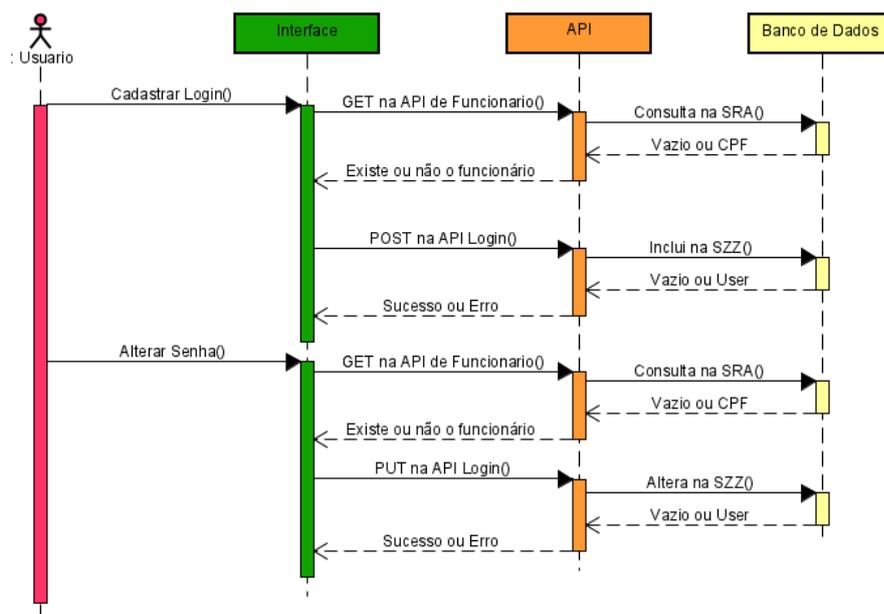


Figura 41 – Diagrama de Sequência de Cadastrar no Usuário e Alterar Senha.

Fonte: elaborado pela autora.

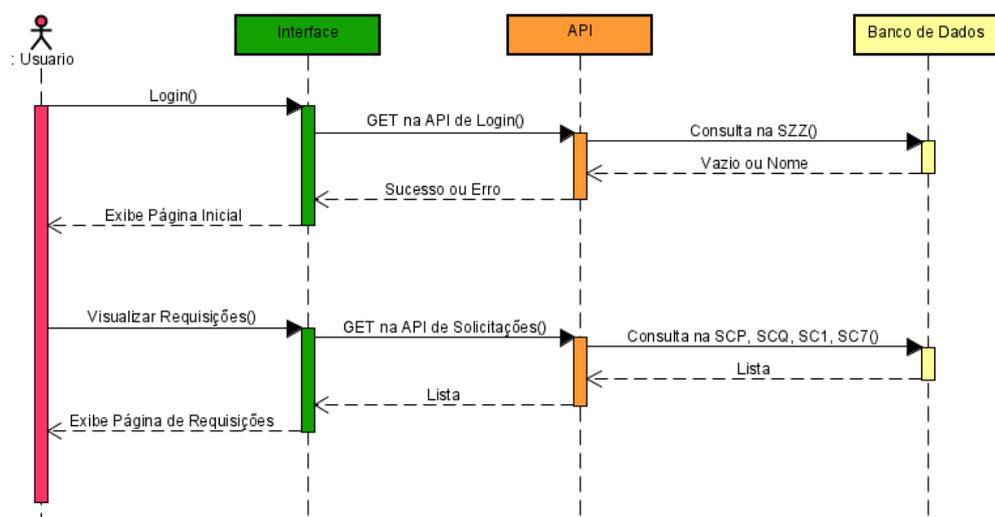


Figura 42 – Diagrama de Sequência de Login e Visualizar Requisições.

Fonte: elaborado pela autora.

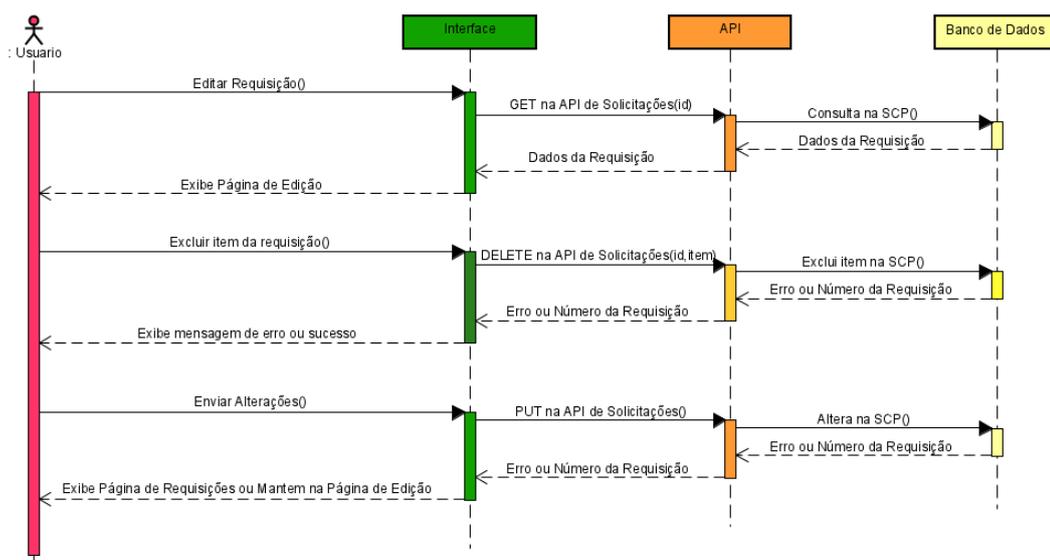


Figura 43 – Diagrama de Sequência de Edição.

Fonte: elaborado pela autora.

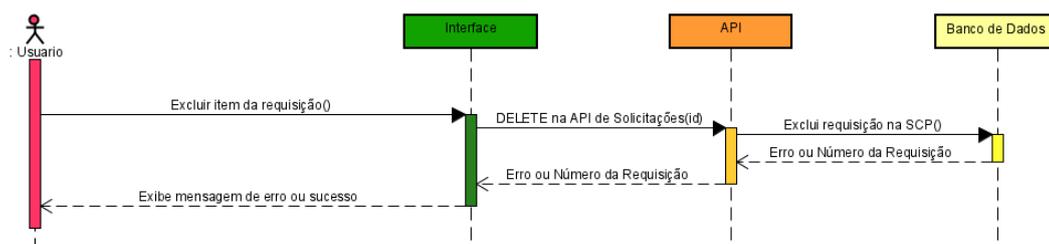


Figura 44 – Diagrama de Sequência de Exclusão.

Fonte: elaborado pela autora.

APÊNDICE B – Modelo Relacional

B.1 Adequações

Para que a aplicação funcione da melhor maneira possível e alcance os seus objetivos objetivos são necessárias algumas personalizações essenciais do sistema, e também algumas sugeridas.

Iniciando pela tabela SCP e SCQ, em que o campo Solicitante é um campo que só permite a seleção dos usuários cadastrados no Protheus, como o objetivo da aplicação é justamente evitar que essas solicitações sejam feitas apenas para os usuários do Protheus, é necessária alguma alteração nestas tabelas.

Visto que o campo Solicitante original do Protheus é um campo não personalizável, é aconselhado que apenas bloqueie o seu uso conforme a [Figura 45](#).

Edição de Campo - CP_SOLICIT

Gerenciador de Base de Dados [Outras Ações] [Cancelar] [Salvar]

Campos: Campo | Informações | Opções | Validações | Uso | Módulos | Regras de dependência

Campo: CP_SOLICIT

Tipo: 1 - Caracter

Tamanho: 25

Decimal: 0

Formato: [Reduzido]

Form. Variavel: [Reduzido]

Contexto: 1 - Real

Propriedade: 2 - Visualizar

TOTS PDV: 2 - Não

Modal: 2 - Não

Grupo de campos: Não tem grupo de campos

Figura 45 – Configuração para bloquear campo Solicitante.

Fonte: elaborado pela autora.

E para registrar os usuários que estão fazendo a requisição incluir dois campos personalizados: O campo Requisitante, que irá armazenar a matrícula do requisitante da aplicação, e o campo virtual, Descrição do Requisitante, que irá apresentar o nome do funcionário que realizou a requisição. A [Figura 46](#) e a [Figura 47](#) mostram as configurações necessárias desses campos.

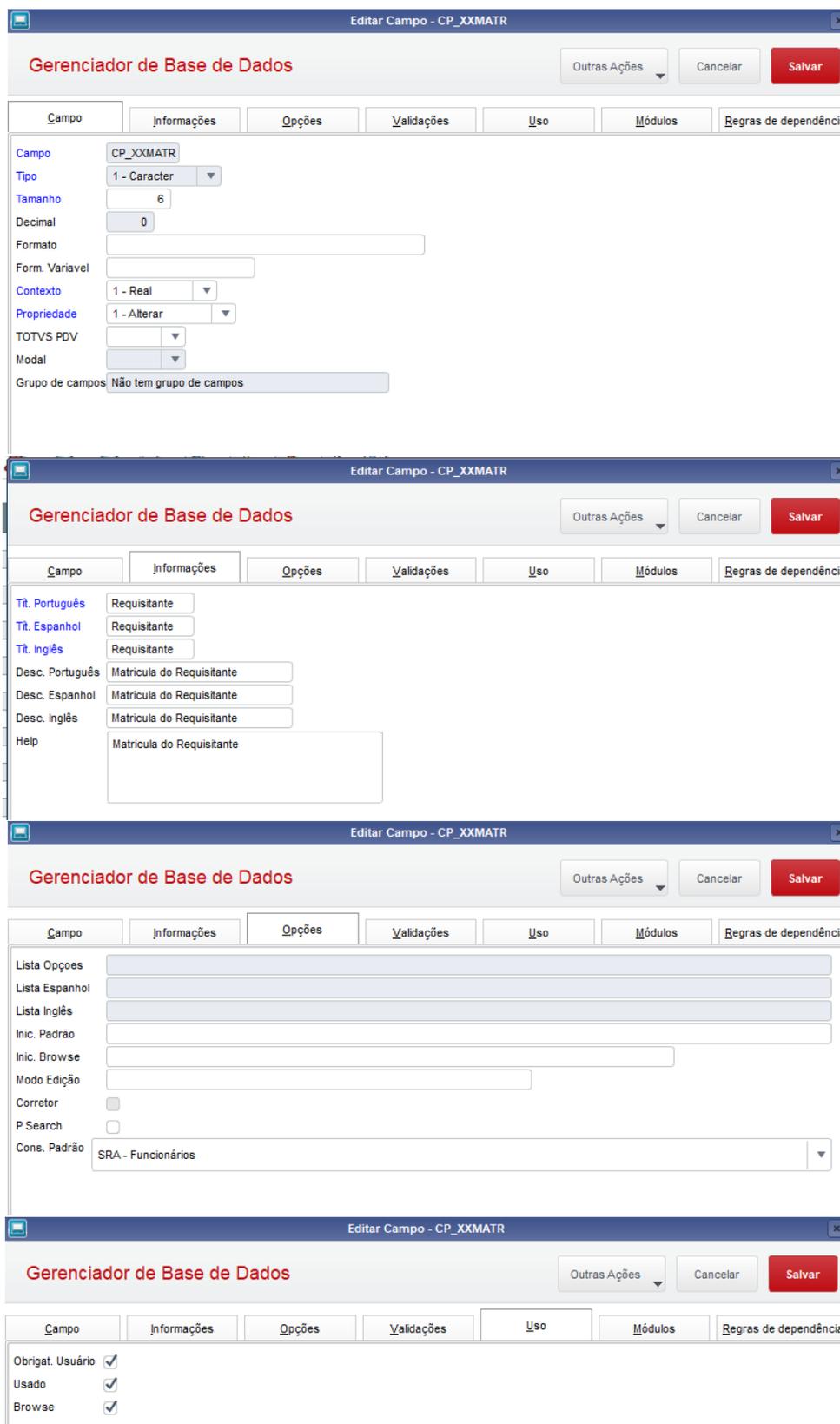


Figura 46 – Criação do campo Requisitante na tabela SCP

Fonte: elaborado pela autora.

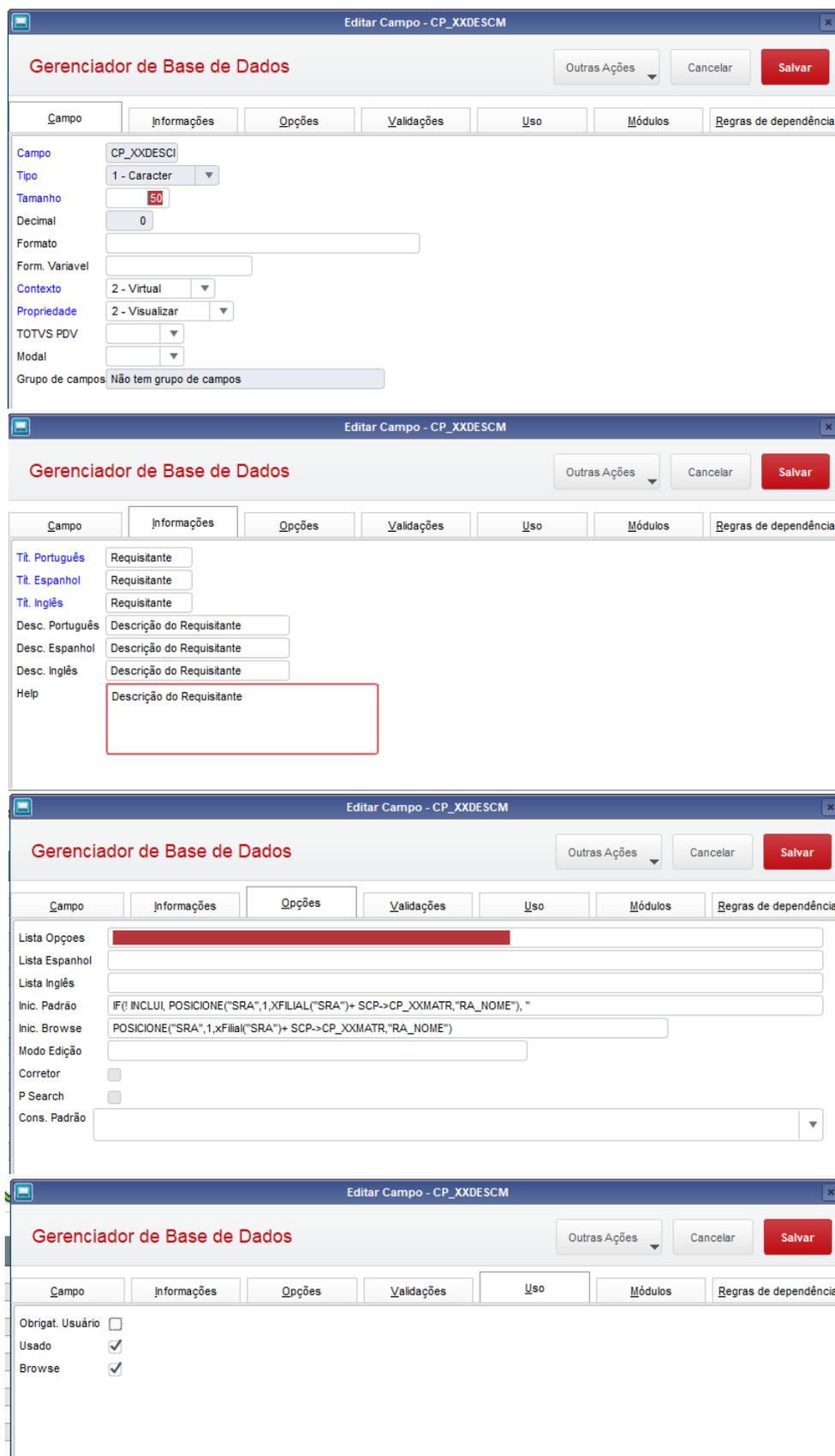


Figura 47 – Criação do campo Descrição do Requisitante na tabela SCP

Fonte: elaborado pela autora.

Além disso, para que o *login* da aplicação *web* possa ser utilizado de forma adequada e segura, sem a necessidade de criar um banco de dados separado para a aplicação é preciso criar uma nova tabela para armazenar estes usuários. Como sugestão foi utilizado o nome SZZ, porém este pode ser alterado conforme o desenvolvedor achar melhor. Os campos necessários nessa tabela são os apresentados na [Figura 48](#).

	-	+	Ordem	Campo	Título	Descrição
	↑	↓	1	ZZ_FILIAL	Filial	Filial do Sistema
	↑	↓	2	ZZ_USER	User	User (CPF)
	↑	↓	3	ZZ_NOME	Nome	Nome
	↑	↓	4	ZZ_SENHA	Senha	Senha Criptografada
	↑	↓	5	ZZ_MAT	Matricula	Matricula
	↑	↓	6	ZZ_MSBLQL	Bloqueado?	Registro bloqueado
	↑	↓	7	ZZ_DTFIM	Data Fim	Data de bloqueio

Figura 48 – Campos da tabela SZZ.

Fonte: elaborado pela autora.

As configurações desses campos podem ser acompanhadas na [Figura 49](#), [Figura 50](#), [Figura 51](#), [Figura 52](#), [Figura 53](#), [Figura 54](#), [Figura 55](#) e [Figura 56](#).

Figura 49 – Campo Filial da tabela SZZ.

Fonte: elaborado pela autora.

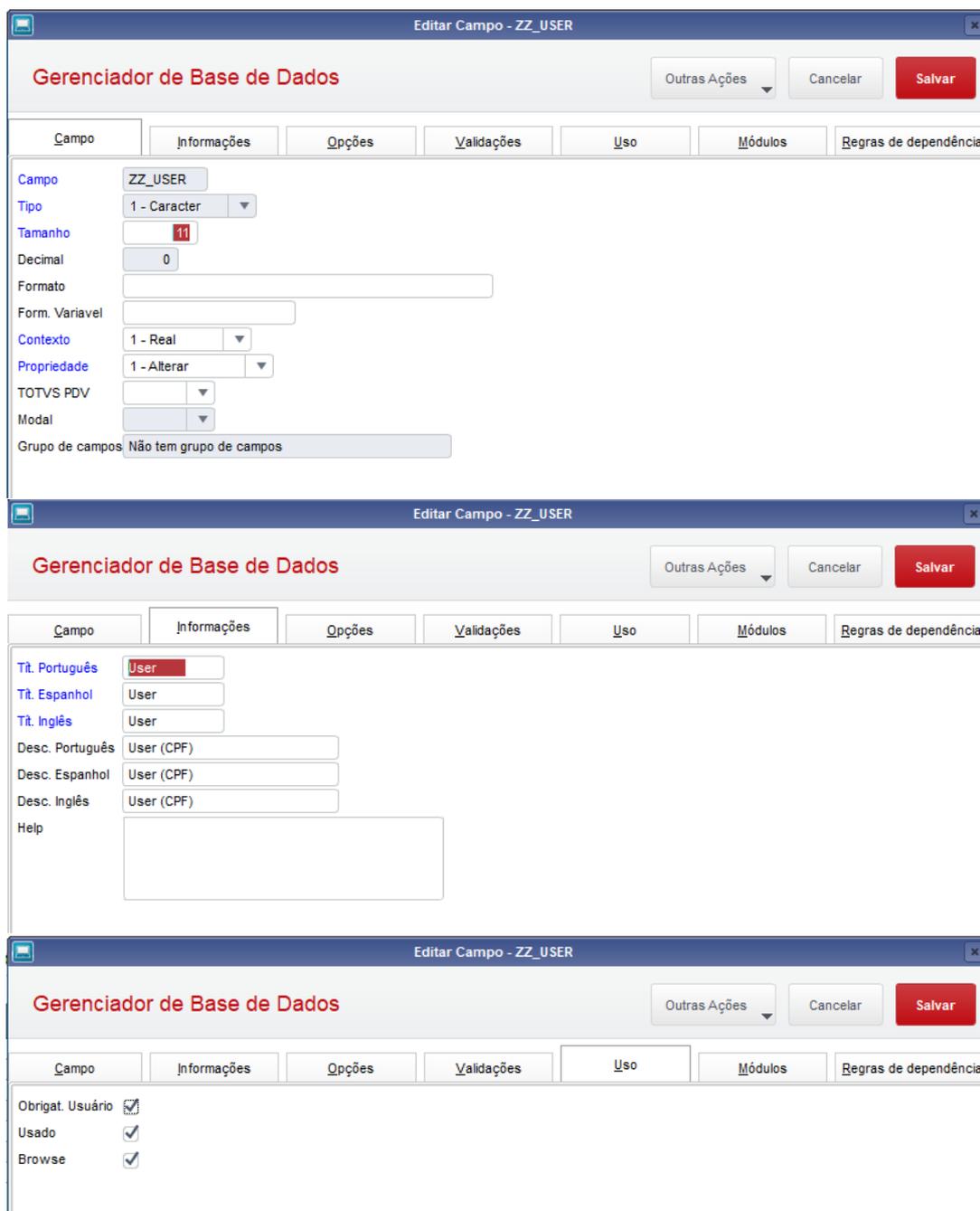


Figura 50 – Campo User da tabela SZZ.

Fonte: elaborado pela autora.

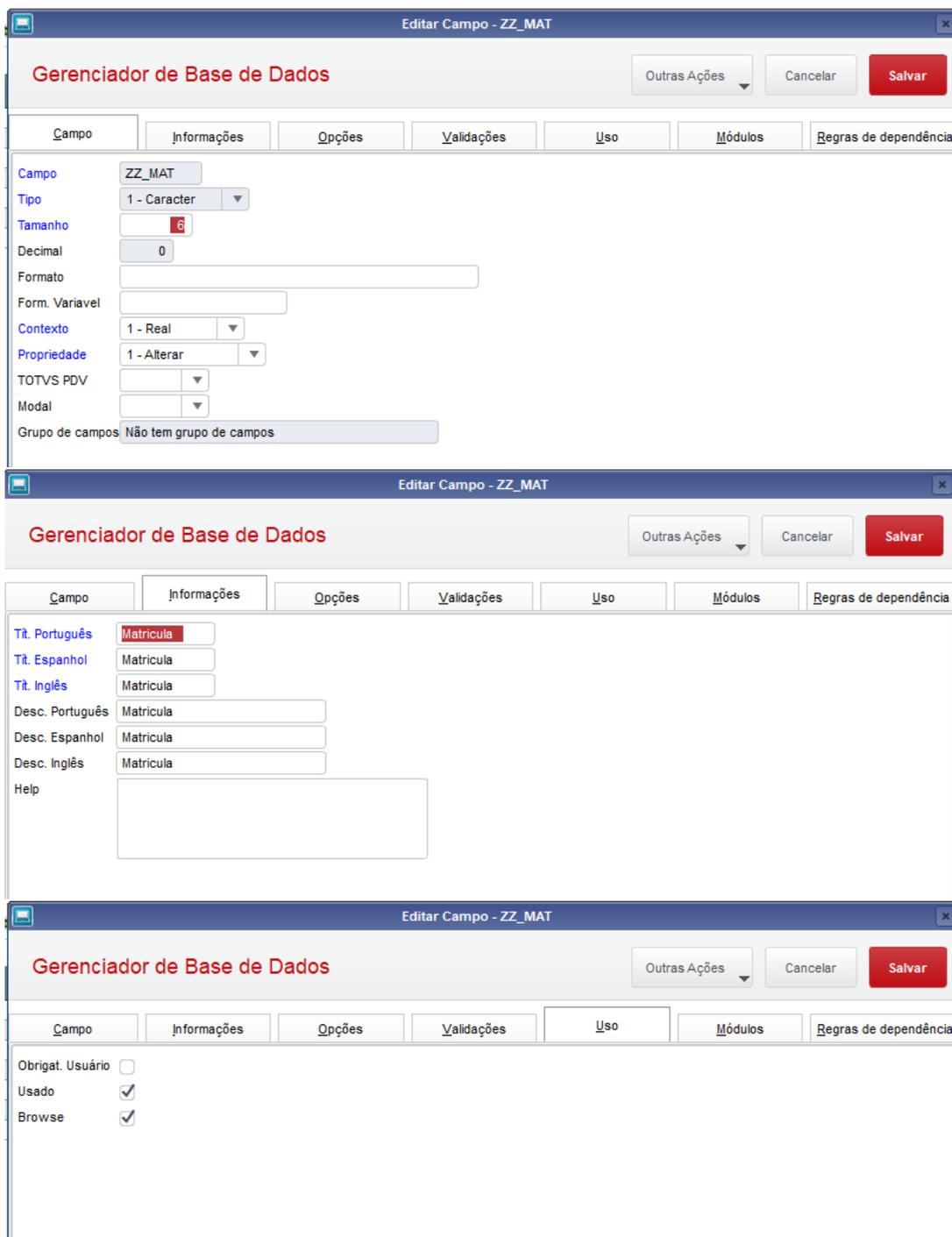


Figura 51 – Campo Matrícula da tabela SZZ.

Fonte: elaborado pela autora.

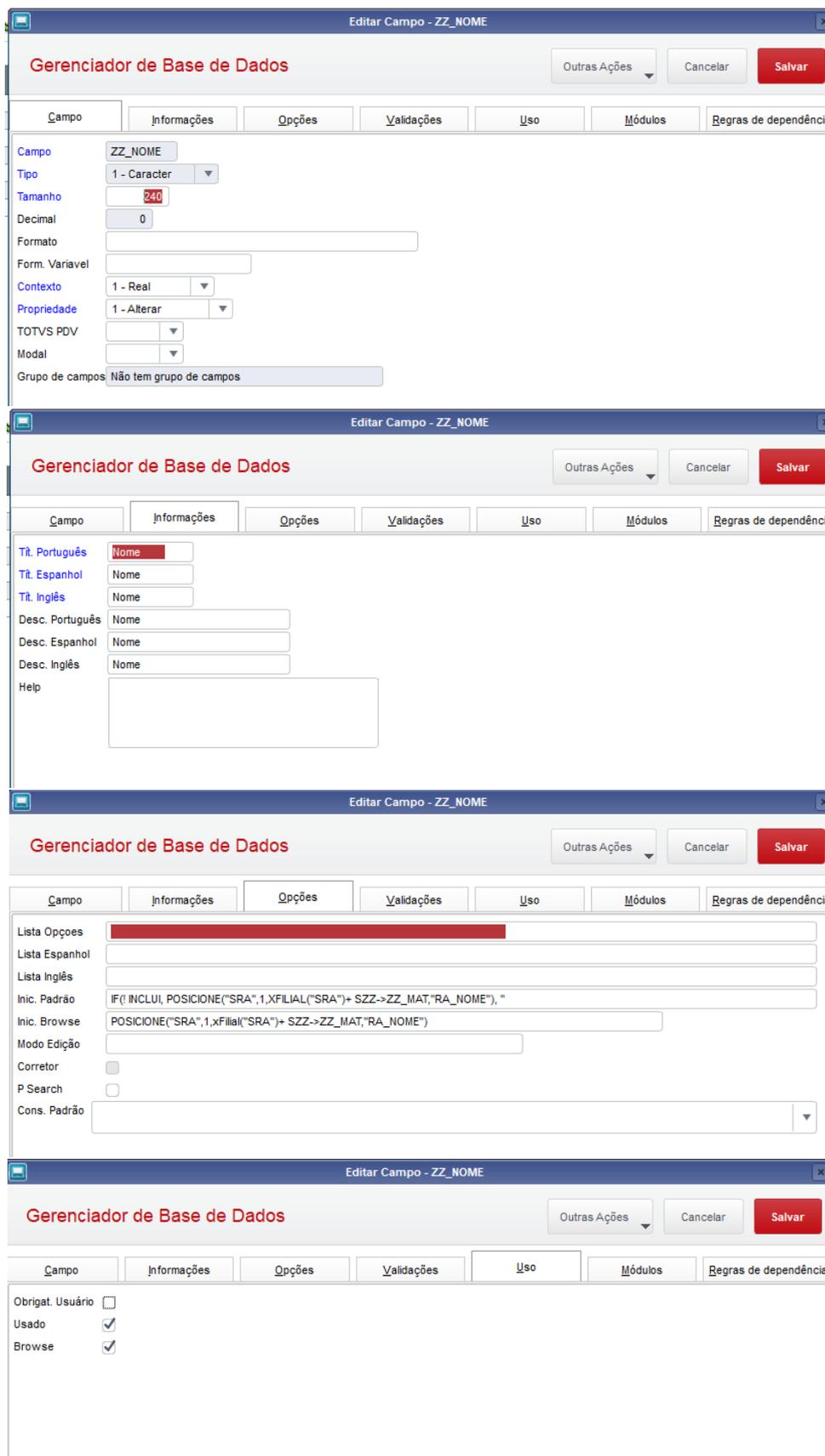


Figura 52 – Campo Nome da tabela SZZ.

Fonte: elaborado pela autora.

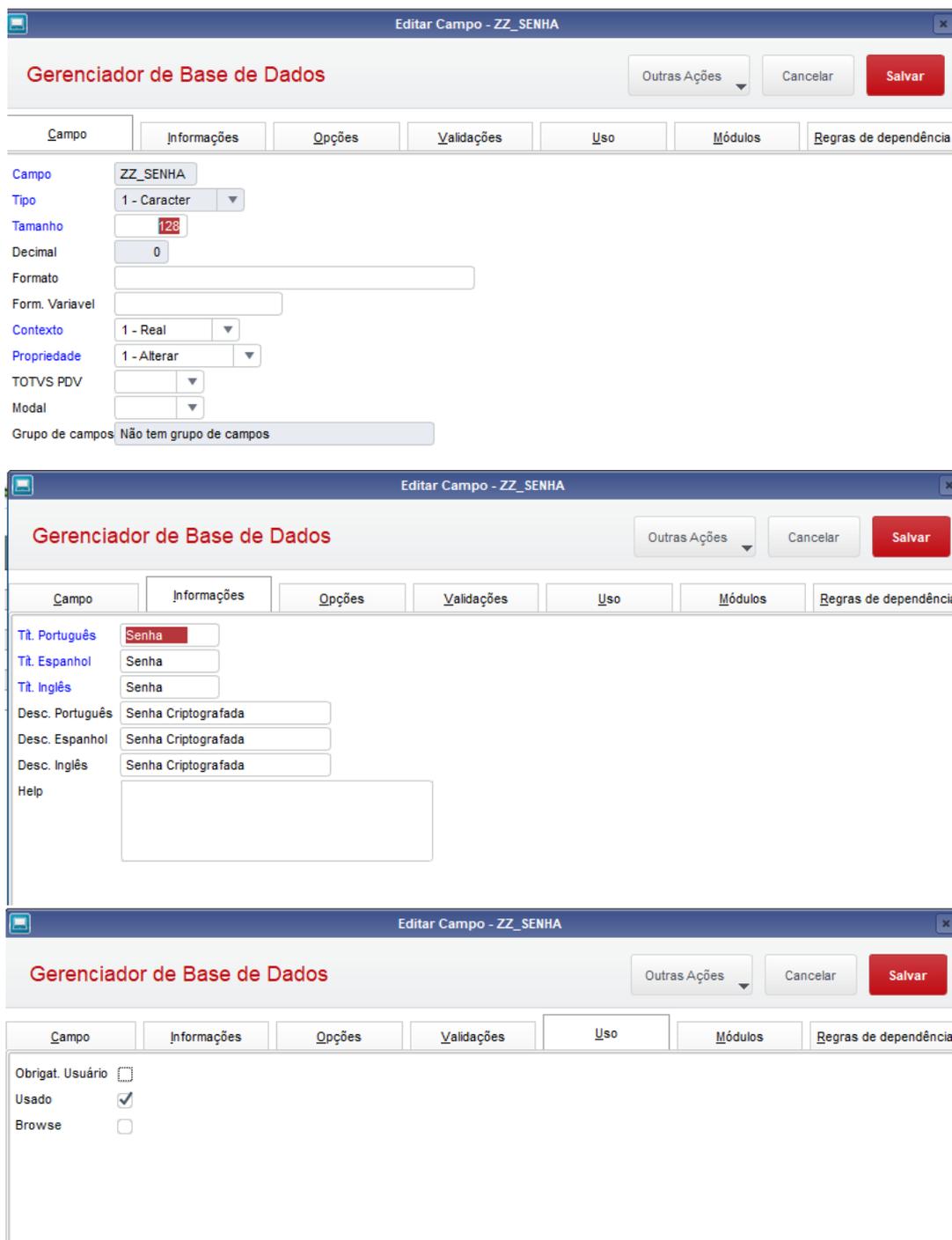


Figura 53 – Campo Senha da tabela SZZ.

Fonte: elaborado pela autora.

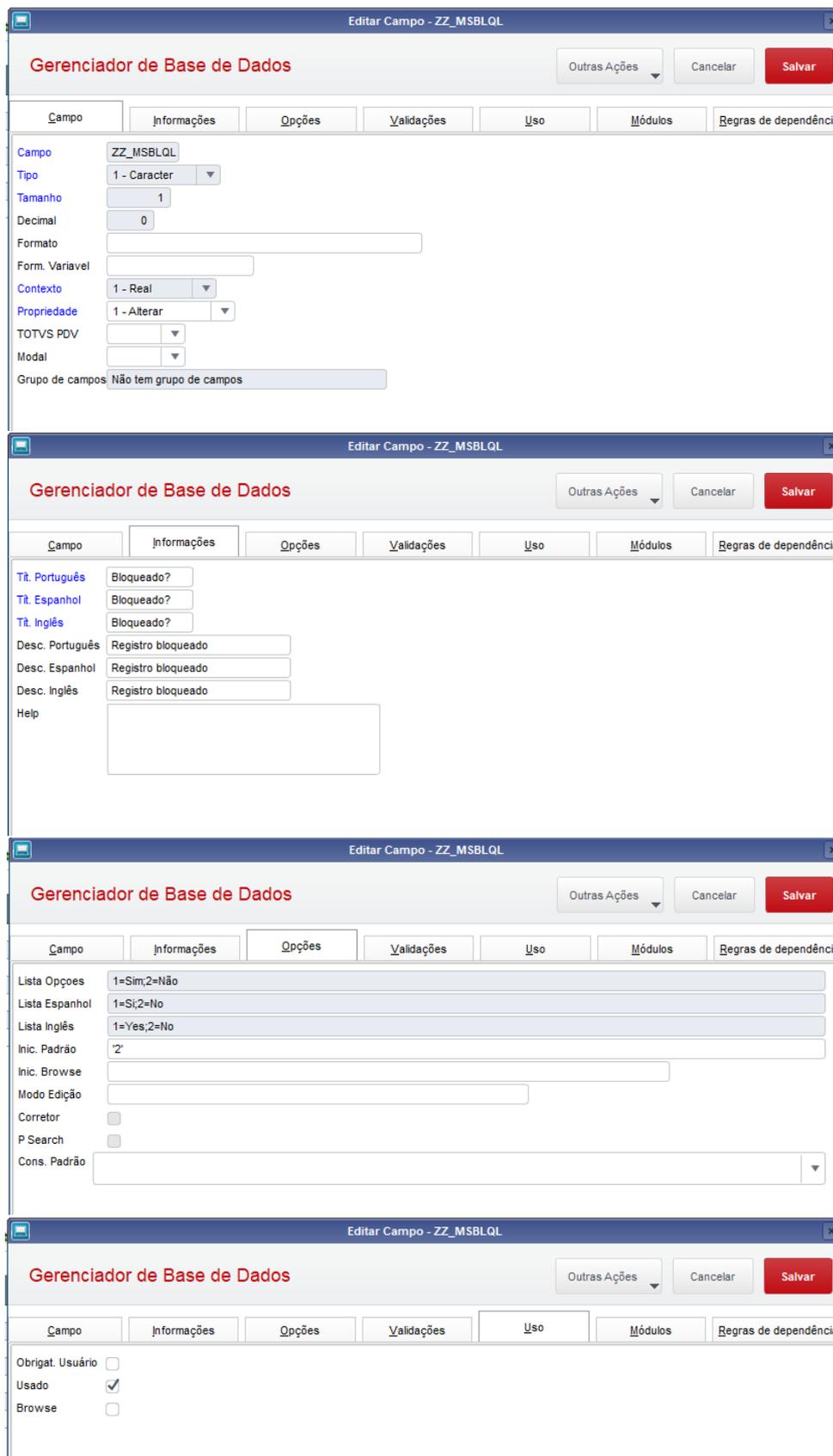


Figura 54 – Campo Bloqueio da tabela SZZ.

Fonte: elaborado pela autora.

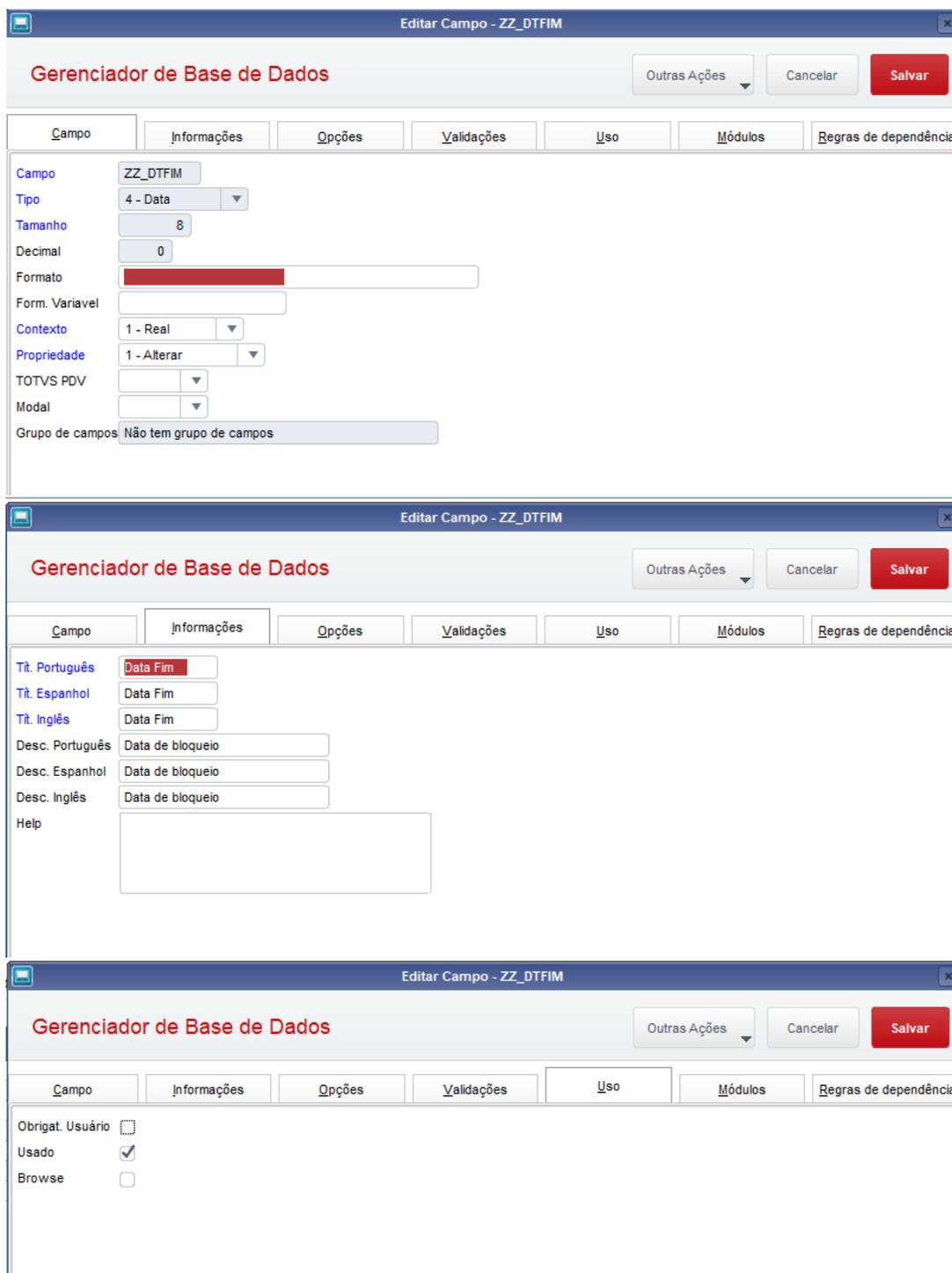


Figura 55 – Campos Data Fim da tabela SZZ.

Fonte: elaborado pela autora.

Ordem	Descrição	Chave
1	Filial + User	ZZ_FILIAL+ZZ_USER

Figura 56 – Índice da tabela SZZ.

Fonte: elaborado pela autora.

Para que seja possível controlar essa tabela de usuários é aconselhada a criação de uma rotina no Protheus, para isso é necessário incluir um novo menu e chamar o fonte ESTSZZ disponível no versionamento do *GitHub*, conforme [Figura 57](#). É sugerido também que o local a ser incluída esta rotina seja o caminho: Estoque/Custos » Atualizações » Movimentações » Internas » Armazém, de modo a centralizar todas as rotinas utilizadas no projeto em um só lugar.

Configuração do item

Desc. Português:

Desc. Espanhol:

Desc. Inglês:

Programa:

Status: Habilitado Desabilitado Inibido

Módulo:

Tipo:

Palavras-Chave Português:

Palavras-Chave Espanhol:

Palavras-Chave Inglês:

Rotinas do browse:

<input checked="" type="checkbox"/>	01
<input checked="" type="checkbox"/>	02
<input checked="" type="checkbox"/>	03
<input checked="" type="checkbox"/>	04
<input checked="" type="checkbox"/>	05
<input checked="" type="checkbox"/>	06
<input checked="" type="checkbox"/>	07
<input checked="" type="checkbox"/>	08
<input checked="" type="checkbox"/>	09
<input checked="" type="checkbox"/>	10

Tabelas:

Alias	Descrição
<input checked="" type="checkbox"/>	A00 Território X Nível do Agrup.
<input type="checkbox"/>	A01 Exceção Territoriais X Agrup.
<input type="checkbox"/>	A02 Exceção Por Níveis do Agrupado
<input type="checkbox"/>	A03 Sequenciador
<input type="checkbox"/>	A04 Sequencia X Agrupador
<input type="checkbox"/>	A05 Script X Rotina
<input type="checkbox"/>	A06 Item Script Validação
<input type="checkbox"/>	A08 Agrup. X Regras Para Pesquisas
<input type="checkbox"/>	A09 Membros X Território
<input type="checkbox"/>	A0A Equipe X Usuarios
<input type="checkbox"/>	A10 Cadastro Conjunto de Etapas
<input type="checkbox"/>	A11 Checklist do Funcionário
<input type="checkbox"/>	A1G Tabela Classificadores Rec.des
<input type="checkbox"/>	A1H Itens Tabela Classif.rec.desp.
<input type="checkbox"/>	A1I Cad.glutinadores Visao Ger.pco
<input type="checkbox"/>	A1J Itens Aglutinadores Visao
<input type="checkbox"/>	AA1 Atendentes

Busca por alias:

Figura 57 – Configuração do menu para incluir rotina de Usuários do Armazém.

Fonte: elaborado pela autora.

Com isso os usuários podem ser cadastrados pela aplicação web e também podem ser cadastrados no Protheus, pelos funcionários do armazém através da rotina Usuários Armazém criada. Porém o principal objetivo desta rotina é o bloqueio de usuários, sempre que um usuário se afastar da empresa, ou não ser mais liberado a fazer requisições ele pode ser bloqueado pelos funcionários do armazém utilizando esta rotina. Basta clicar em Alterar, mudar o Status de Bloqueio para 1 e inserir a data de bloqueio, conforme [Figura 58](#).

TOTVS | Estoque/Custos

TOTVS Construção e Projetos POSTGRES Bela Administrador 21/10/2022 Teste / Matrç

Cadastro de Usuários Armazém Web - ALTERAR

Outras Ações Cancelar Salvar

User * 12512405699 Nome EMILY BRITO Senha 81dc8bdb52d04dc20036db0313e0055

Matricula 000002 Bloqueado? 1 - Sim Data Fim 21/10/2022

Figura 58 – Exemplo de bloqueio de usuário do armazém pela rotina criada no Protheus.

Fonte: elaborado pela autora.

Por fim, na rotina de Gerar Pré-Requisição, ao clicar em Gerar, é apresentada a janela mostrada na [Figura 59](#), e ao clicar em Param. é mostrada a [Figura 60](#). Nela o funcionário do armazém que estará atendendo a requisição pode definir os parâmetros conforme a política da empresa, porém para alcançar todo o potencial deste trabalho e gerar as solicitações de compra de forma automática é necessário que o campo Gera Sol. Compras/Aut.Entrega esteja com Sim selecionado.

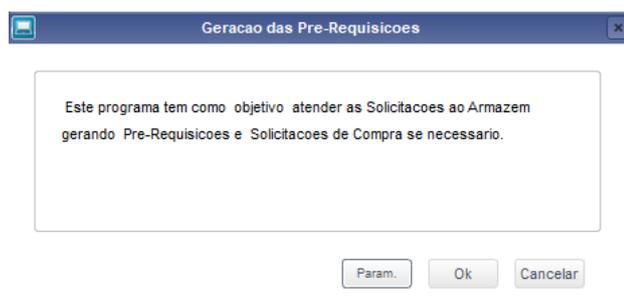


Figura 59 – Janela de confirmação para Gerar Pré-Requisito no Protheus.

Fonte: elaborado pela autora.

The image displays two screenshots of the 'Parametros' dialog box in the Protheus system. The top screenshot shows the following parameters: 'Considera Data da ?' set to 'Emissao', 'Cons. o Sld Prev. de Entrada ?' set to 'Nao', 'Gera Sol.Compras/Aut.Entrega ?' set to 'Sim', 'Cons. o Armazem da SA ?' set to 'Nao', 'Saldo do Armazem ?' is an empty text box, and 'Saldo Ate o Armazem ?' is set to '99'. The bottom screenshot shows: 'Saldo do Armazem ?' is an empty text box, 'Saldo Ate o Armazem ?' is set to '99', 'Considera Lote Economico ?' set to 'Nao', 'Avalia empenhos para OP ?' set to 'Nao', 'Aglutina documentos gerados ?' set to 'Sim por SA', and 'Subtrai estoque de segurança ?' set to 'Nao'. Both screenshots include 'Cancelar' and 'OK' buttons at the bottom right.

Figura 60 – Janela de definição de parâmetros para a Geração de Pré-Requisição no Protheus.

Fonte: elaborado pela autora.

Ao acessar o caminho Atualizações » Movimentações » Internas » Armazém os menus estarão apresentados conforme a [Figura 61](#). Nela é possível observar a existência da rotina Liberar Solicitação, que serve para definir que uma solicitação só poderá gerar pré-requisição quando for liberada. Para ser ativada é necessária a ativação do parâmetro MV_APROVSA. Esta rotina seria interessante para empresas em que a gerência ou diretoria precisa autorizar a requisição do funcionário, ou a requisição de certos produtos, porém esses direcionamentos e limitações precisariam ser personalizadas no sistema através dos pontos de entrada. Como o objetivo deste trabalho é implementar o padrão, essa rotina não é ativada e não é considerada as liberações, porém é apresentada como ponto de melhoria para empresas que se enquadram nessas situações.

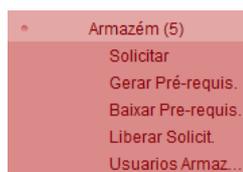


Figura 61 – Rotinas no menu do Protheus.

Fonte: elaborado pela autora.