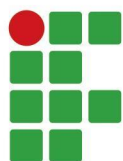


INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA FARROUPILHA
CAMPUS AVANÇADO URUGUAIANA
CURSO TÉCNICO EM INFORMÁTICA INTEGRADO



**INSTITUTO
FEDERAL**

Farroupilha

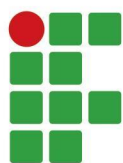
Campus Avançado
Uruguaiana

ANTHONY RODRIGUES DA SILVA

HELP.COM

Uruguaiana/RS

2021/2



**INSTITUTO
FEDERAL**

Farroupilha

Campus Avançado
Uruguaiana

ANTHONY RODRIGUES DA SILVA

HELP.COM

Relatório referente ao Trabalho de Conclusão de Curso, apresentado como requisito para obtenção do título de Técnico em Informática, do Instituto Federal de Educação, Ciência e Tecnologia Farroupilha Campus Avançado Uruguaiana.

Orientador(a): **Thiago Cassio Krug**

Uruguaiana/RS

2021/2

ANTHONY RODRIGUES DA SILVA

HELP.COM

Relatório referente ao Trabalho de Conclusão de Curso, apresentado como requisito para obtenção do título de Técnico em Informática, do Instituto Federal de Educação, Ciência e Tecnologia Farroupilha Câmpus Avançado Uruguaiana.

Aprovado em 17 de dezembro de 2021.

BANCA EXAMINADORA

Thiago Cassio Krug

Leandro Martins Dallanora

Michel Michelin

Sumário

1 INTRODUÇÃO	5
2 OBJETIVOS	7
2.1 OBJETIVO GERAL	7
2.2 OBJETIVOS ESPECÍFICOS	7
3 REVISÃO BIBLIOGRÁFICA	8
3.1 SISTEMAS SEMELHANTES	8
3.2 CONCLUSÕES	9
3.3 CONCEITOS TÉCNICOS	9
4 METODOLOGIA	10
5 DESENVOLVIMENTO DO SISTEMA	11
5.1 Documentação de requisitos	11
5.1.1 Convenções, termos e abreviações	11
5.1.2 Prioridades dos requisitos	11
5.1.3 Atores do Sistema	12
5.1.4 Requisitos funcionais do sistema	12
5.2 Casos de uso	14
5.2.1 Diagrama de casos de uso	14
5.3 Casos de uso expandidos	16
5.4 Diagrama do Banco de Dados	20
5.5 Interfaces do sistema	20
6 CONSIDERAÇÕES FINAIS	24
7 REFERÊNCIAS	25

1 INTRODUÇÃO

A existência de diversas linguagens de programação, bem como a vasta gama de comandos existentes em cada uma delas, faz com que a memorização das mesmas, por parte dos programadores, seja quase impossível, além de um tanto quanto desnecessária. Tendo em vista que nem todas são utilizadas na criação de programas. Além do fato que, para programadores “iniciantes” essa vasta quantidade de possibilidades pode gerar confusões.

Dito isso, a ideia de meu trabalho de conclusão de curso (TCC) é desenvolver um site para o armazenamento colaborativo de estruturas de linguagens de programação.

Na prática, o site vai funcionar como uma biblioteca de comandos, divididos por linguagens e funções, e, como ferramenta de estudo, podendo assim não só auxiliar o usuário na parte de programação, mas também na parte do aprendizado.

Ele irá seguir o processo de desenvolvimento de software cascata, que compreende as etapas sequências de levantamento de requisitos, criação do projeto do software bem como do banco de dados, desenvolvimento do sistema, e a realização dos testes das funcionalidades.

Inicialmente, ele conterá apenas algumas das linguagens e comandos mais usados atualmente, como alguns exemplos de PHP, HTML, CSS e Javascript, porém, por conta de seu sistema colaborativo, novas linguagens poderão ser adicionadas pelos usuários, assim como linguagens que já estejam no site poderão ser editadas pelos mesmos, porém, antes de serem incorporadas ao site, todas as informações passaram por um processo de avaliação, necessário para manter a qualidade das informações.

Deste ponto em diante o restante deste documento está organizado da seguinte forma:

O capítulo 2 apresenta os objetivos do projeto, tanto os gerais quanto os específicos.

O capítulo 3 é uma revisão bibliográfica, apresentando conceitos técnicos , sistemas semelhantes ao meu e a conclusão do projeto.

O capítulo 4 apresenta a metodologia, detalhando os objetivos específicos do projeto.

O capítulo 5 apresenta o desenvolvimento do sistema.

Por fim, o capítulo 6 apresenta as considerações finais deste trabalho.

2 OBJETIVOS

2.1 OBJETIVO GERAL

Desenvolver um sistema que armazene informações sobre linguagens de programação, para auxiliar os desenvolvedores na criação de seus programas.

2.2 OBJETIVOS ESPECÍFICOS

1. Levantar os requisitos do sistema;
2. Aprofundar conhecimentos em banco de dados, HTML e PHP;
3. Elaborar o diagrama de casos de uso e os casos de uso expandidos;
4. Criar o Diagrama Entidade-Relacionamento baseado nos requisitos elicitados;
5. Desenvolver o sistema;
6. Testar as funcionalidades do sistema;

3 REVISÃO BIBLIOGRÁFICA

Esta seção apresenta trabalhos relacionados e um estudo mais aprofundado de cada um deles, a fim de que se tenha uma base mais desenvolvida do tema, validando também a importância do mesmo. Abaixo, na subseção 3.1 encontram-se os sistemas semelhantes e na subseção 3.2 analisa os trabalhos, artigos e sites apresentados. A última subseção, 3.3, apresenta as ferramentas que serão utilizadas para desenvolver este trabalho.

3.1 SISTEMAS SEMELHANTES

A ideia de armazenar dados sobre determinados assuntos é uma ideia já explorada, exemplos de sites semelhantes que podem ser tidos são:

A Wikipédia, que é um projeto de enciclopédia multilíngue de licença livre, baseado na web e escrito de maneira colaborativa. O projeto encontra-se sob administração da Fundação Wikipédia, uma organização sem fins lucrativos cuja missão é “empoderar e engajar pessoas pelo mundo para coletar e desenvolver conteúdo educacional sob uma licença livre ou no domínio público”.

O PHP.net, que é um site especializado em linguagem php, trazendo desde informações sobre suas versões mais antiga (como a versão 3.0 lançada em 06-jun-1998) até as mais novas (como a versão 8.0.7 lançada em 03-jun-2021), o site também informa sobre correção de erros na linguagem além de conter uma biblioteca de comandos de cada versão.

E o Brainly que é uma empresa de tecnologia educacional localizada em Cracóvia, Polônia, que gera um grupo de redes sociais de aprendizagem para estudantes e educadores. Está disponível em 13 versões linguísticas que são visitadas por mais de 40 milhões de usuários únicos mensais de mais de 35 países. A primeira versão do grupo foi lançada em 2009 na Polônia sob o nome brainly.pl O sistema da plataforma é baseado em elementos de gamificação, como, por exemplo, pontos que permitem os usuários fazerem e responder perguntas de matérias escolares. O objetivo do Brainly é promover a aprendizagem colaborativa online aproveitando as novas tecnologias.

3.2 CONCLUSÕES

Ter um site colaborativo para armazenamento de estruturas de linguagens de programação, pode não só auxiliar programadores veteranos tirando alguma possível dúvida poção ter, como também pode servir como uma ferramenta de estudo para diversas linguagens de programação, e o fato do usuário poder editar as informações contidas nele, ajuda ainda mais no controle de qualidade das mesmas, podendo assim torná-lo uma ferramenta confiável e segura.

3.3 CONCEITOS TÉCNICOS

Abaixo apresento os conceitos e ferramentas que serão utilizados durante o desenvolvimento deste trabalho:

HTML: Linguagem de marcação utilizada na criação de web-sites.

PHP: Linguagem de programação open-source muito utilizada especialmente no desenvolvimento web, uma vez que pode ser mesclado com a linguagem de marcação HTML.

JavaScript: Linguagem de programação que permite o desenvolvedor criar conteúdo dinâmico, controlar multimídias, imagens animadas, entre outros.

WAMP: É um pacote de programas que facilita a configuração de um software interpretador de scripts local e um banco de dados.

4 METODOLOGIA

A tabela a seguir (Tabela 1) apresenta os procedimentos metodológicos deste trabalho de conclusão de curso.

Tabela: Metodologia

Objetivo Específico	Ação
1. Levantar os requisitos do sistema.	<ul style="list-style-type: none">● Coletar requisitos através de pesquisas em sites semelhantes.
2. Aprofundar conhecimentos em banco de dados, HTML e PHP;	<ul style="list-style-type: none">● Estudar banco de dados, HTML e PHP.
3. Elaborar o diagrama de casos de uso e os casos de uso expandidos;	<ul style="list-style-type: none">● Criar o diagrama de caso de uso e o diagrama de atividades.● Fazer os casos de uso expandidos.
4. Criar o Diagrama Entidade-Relacionamento baseado nos requisitos listados.	<ul style="list-style-type: none">● Fazer o diagrama de Entidade-Relacionamento.● Criar o modelo de banco de dados.
5. Desenvolver o sistema;	<ul style="list-style-type: none">● Desenvolver o sistema com base nos diagramas.● Desenvolver o sistema usando HTML e linguagem de programação PHP.
6. Testar as funcionalidades do sistema de orientação de TCC.	<ul style="list-style-type: none">● Realizar testes de para verificar se está realizando o necessário

5 DESENVOLVIMENTO DO SISTEMA

Esta seção apresenta o processo do desenvolvimento do sistema proposto neste trabalho. Ela está dividida em 5 partes: Documentação de requisitos, Casos de uso, Diagrama de dados, Cronograma e Interfaces do sistema.

5.1 DOCUMENTAÇÃO DE REQUISITOS

Esta seção especifica os requisitos do sistema, que fornecem ao desenvolvedor as informações necessárias para a implementação do sistema.

5.1.1 Convenções, termos e abreviações

Por convenção, a referência a requisitos é feita através do nome da subseção onde eles estão descritos, seguidos do identificador do requisito, de acordo com a especificação a seguir:

[identificador do requisito (RN-xx) – nome do requisito]

Os requisitos devem ser identificados com um identificador único. A numeração inicia com o identificador [RN-01] e prossegue sendo incrementada à medida que forem surgindo novos requisitos.

5.1.2 Prioridades dos requisitos

Para estabelecer a prioridade dos requisitos foram adotadas as denominações “essencial”, “importante” e “desejável”.

- **Essencial:** é o requisito sem o qual o sistema não entra em funcionamento. São requisitos imprescindíveis, que devem ser implementados impreterivelmente.

- **Importante:** é o requisito sem o qual o sistema entra em funcionamento, mas de forma não satisfatória. Requisitos importantes devem ser implementados, mas, se não forem, o sistema poderá ser implantado e usado mesmo assim.

- **Desejável:** é o requisito que não compromete as funcionalidades básicas do sistema, isto é, o sistema pode funcionar de forma satisfatória sem ele. Requisitos

desejáveis podem ser deixados para versões posteriores do sistema, caso não haja tempo hábil para implementá-los na versão que está sendo especificada.

5.1.3 Atores do Sistema

O sistema apresenta dois atores, o usuário, que tem acesso a funções básicas, e o administrador, que tem acesso a todas as funções do sistema.

- **Usuário:** O usuário do sistema, detém as funcionalidades: se cadastrar no sistema, se logar e cadastrar novas funções.
- **Administrador:** O administrador possui todas as funcionalidades listadas no usuário com a adição das funções: excluir funções e excluir usuários.

5.1.4 Requisitos funcionais

Abaixo são listados os requisitos funcionais previstos para o sistema, que foram extraídos com ajuda do orientador e a partir de sugestões de colegas.

[RF-01] Manter conta

Descrição do RF: permite ao usuário controle total de sua conta no sistema, desde seu cadastro e gerenciamento até sua exclusão.

Prioridade: Essencial

Entradas e pré-condições: Usuário sem cadastro ou logado no sistema.

Saídas e pós condições: Usuário cadastrado no sistema, conta alterada ou excluída.

[RF-02] Realizar login

Descrição do RF: Permite que o usuário efetue o login no sistema.

Prioridade: Essencial

Entradas e pré-condições: Estar cadastrado no sistema.

Saídas e pós condições: Usuário entra no sistema.

[RF-03] Pesquisar Comandos e Funções

Descrição do RF: Permite que o usuário pesquise comandos e funções no sistema

Prioridade: Essencial

Entradas e pré-condições: Estar no sistema.

Saídas e pós condições: Usuário entra no sistema.

[RF-04] Adicionar Comandos e Funções

Descrição do RF: Permite que o usuário Adicione comandos e funções no sistema

Prioridade: Essencial

Entradas e pré-condições: Estar logado no sistema.

Saídas e pós condições: O sistema fará a inclusão dos comandos e funções solicitadas pelo usuário.

[RF-05] Recuperar senha

Descrição do RF: Permite que o usuário recupere sua conta via email.

Prioridade: Importante

Entradas e pré-condições: Estar cadastrado no sistema.

Saídas e pós condições: O usuário com a conta recuperada.

[RF-06] Manter Adições feitas pelo usuário

Descrição do RF: Permite que o usuário tenha suas adições mantida no sistema

Prioridade: Essencial

Entradas e pré-condições: Estar cadastrado no sistema.

Saídas e pós condições: Adições registradas

[RF-07] Excluir Comandos e Funções

Descrição do RF: Permite que o administrador exclua comandos e funções adicionados no sistema

Prioridade: Importante:

Entradas e pré-condições: Estar logado no sistema.

Saídas e pós condições: O sistema fará a exclusão dos comandos e funções solicitadas pelo administrador.

[RF-07] Excluir Usuários

Descrição do RF: Permite que o administrador exclua usuários do sistema

Prioridade: Importante:

Entradas e pré-condições: Estar logado no sistema.

Saídas e pós condições: O sistema fará a exclusão dos usuários solicitados pelo administrador.

5.2 CASOS DE USO

5.2.1 Diagrama de casos de uso

A figura 1 apresenta o diagrama de casos de uso desenvolvido para o usuário do sistema. Nele podemos ver que o ator do sistema é o usuário, que poderá criar sua própria conta no sistema, assim acessando as demais funcionalidades.

A figura 2 apresenta o diagrama de casos de uso desenvolvido para o administrador do sistema. Nele podemos ver que o ator do sistema é o administrador, que poderá além de ter acesso às mesmas funções do usuário (listadas no diagrama do usuário) também terá acesso às funções excluir conta e excluir usuário.

Figura 1: Usuário

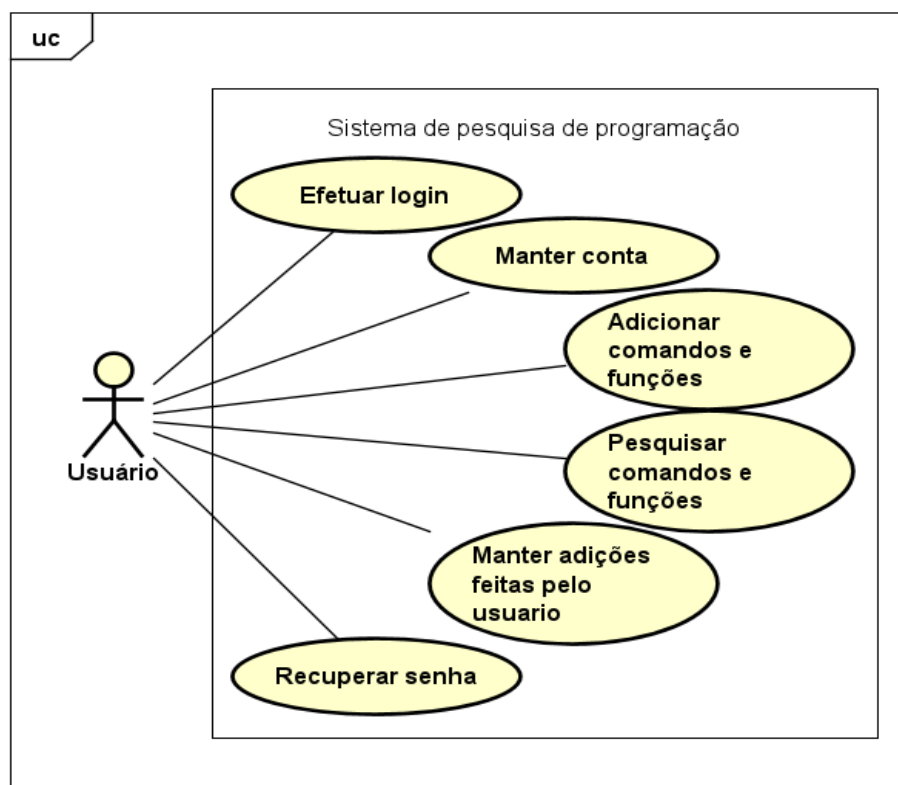
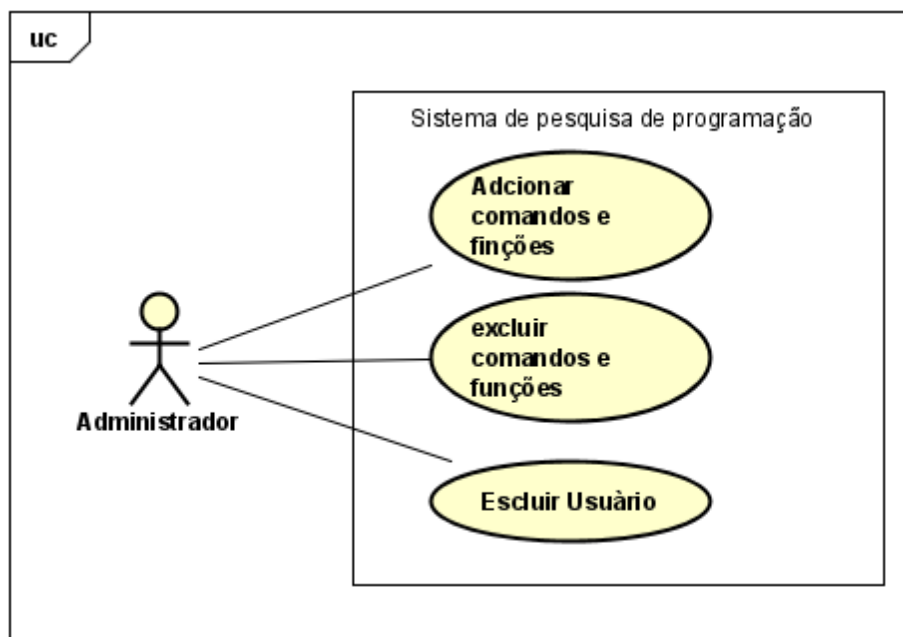


Figura 2: Administrador



5.2.2 Casos de Uso Expandidos

Abaixo seguem as especificações para cada caso de uso (tabelas 1 a 7).

Tabela 1: Efetuar login

Nome	Efetuar login
Ator	Usuário e Administrador
Pré-condições	Ter uma conta cadastrada no sistema
Fluxo principal	
1) O usuário solicita o formulário de login. 2) O sistema exibe o formulário de login. 3) O usuário preenche o formulário e solicita o login no sistema. 4) O sistema verifica e realiza a autenticação dos dados e encaminha o usuário para a página do usuário ou do administrador	
Fluxo Alternativo	
Fluxo de Execução	
a) O sistema verifica que os dados estão incorretos e notifica o usuário, para que o mesmo possa corrigi-los.	

Tabela 2: Manter Conta

Nome	Manter Conta
Ator	Usuário e Administrador
Pré-condições	Nenhuma
Fluxo principal	
<p>- Cadastrar conta</p> <ol style="list-style-type: none"> 1) O usuário solicita o formulário de cadastro. 2) O sistema exibe o formulário de cadastro. 3) O usuário preenche os campos do formulário e solicita o registro das informações. 4) O sistema verifica os dados, salva as informações e encaminha o usuário para a página de login. <p>- Excluir conta</p> <ol style="list-style-type: none"> 1) O administrador solicita o formulário de gerenciamento de conta. 2) O sistema exibe o formulário com as informações do perfil do usuário. 3) O administrador solicita a exclusão da conta. 4) O sistema remove todos os dados, vinculados ao usuário. 	
Fluxo Alternativo	
<ol style="list-style-type: none"> a) O administrador não confirma a exclusão da conta. b) O sistema cancela a operação de exclusão da conta. 	
Fluxo de Execução	

Tabela 3: Adicionar Comandos e Funções

Nome	Adicionar Comandos e Funções
Ator	Usuário e Administrador
Pré-condições	Ter uma conta cadastrada no sistema. Estar logado no sistema
Fluxo principal	
<p>- Cadastrar ou alteração de Comandos ou Função</p> <ol style="list-style-type: none"> 1) O usuário seleciona a função após pesquisá-la, ou, a adiciona no formulário. 2) O sistema exibe o formulário de cadastro de comandos ou funções. 3) O usuário preenche o formulário e solicita o registro ou atualização da função ou comando. 4) O sistema verifica os dados, salva as informações e encaminha o usuário para a página principal, com a alteração ou adição já registrada. 	
Fluxo Alternativo	
<ol style="list-style-type: none"> a) O usuário não confirma a adição ou atualização. b) O sistema cancela a operação de atualização ou adição. 	
Fluxo de Execução	

--

Tabela 4: Pesquisar Comandos e Funções

Nome	Pesquisar Comandos e Funções
Ator	Usuário e Administrador
Pré-condições	Nenhum
Fluxo principal	
<p>-Pesquisar comando ou Função</p> <p>1) O usuário realiza a pesquisa do comando ou função que deseja.</p> <p>2) O sistema exibe o resultado pesquisado, com todas as informações cadastradas para o usuário.</p> <p>3) O usuário visualiza as informações.</p>	
Fluxo Alternativo	
<p>a) O sistema não exibe o resultado pesquisado, pois as informações ainda não foram adicionadas no sistema.</p> <p>b) O usuário adiciona informações ao resultado de sua pesquisa.</p>	
Fluxo de Execução	

Tabela 5: Excluir Função

Nome	Excluir Função
Ator	Administrador
Pré-condições	Ter uma conta cadastrada no sistema. Estar logado no sistema
Fluxo Principal	
<p>- Excluir Comandos ou Função</p> <p>1) O Administrador vai a página de gerenciamento de funções.</p> <p>2) O Administrador seleciona o comando ou a função a qual deseja excluir.</p> <p>3) O sistema apaga o comando ou a função.</p>	
Fluxo Alternativo	
<p>a) O Administrador seleciona o comando ou a função a qual deseja excluir.</p> <p>b) O Administrador não apaga o comando ou a função.</p> <p>c) O sistema não apaga o comando ou a função.</p>	
Fluxo de Execução	

Tabela 6: Excluir Usuário

Nome	Excluir Função
Ator	Administrador
Pré-condições	Ter uma conta cadastrada no sistema. Estar logado no sistema
Fluxo Principal	
<p>- Excluir Usuário</p> <p>1) O Administrador vai a página de gerenciamento de usuário.</p> <p>2) O Administrador seleciona o usuário a qual deseja excluir.</p> <p>3) O sistema apaga o usuário.</p>	
Fluxo Alternativo	
<p>a) O Administrador seleciona o usuário a qual deseja excluir.</p> <p>b) O administrador não apaga o usuário.</p> <p>c) O sistema não apaga o usuário.</p>	
Fluxo de Execução	

Tabela 7: Recuperar Senha

Nome	Recuperar Senha
Ator	Usuário e Administrador
Pré-condições	Ter uma conta cadastrada no sistema.
Fluxo principal	
<p>1) O usuário solicita o formulário de login.</p> <p>2) O sistema exibe o formulário de login.</p> <p>3) O usuário clica na opção “Esqueci minha senha”.</p> <p>4) O sistema exibe o formulário de recuperação de conta.</p> <p>5) O usuário preenche o formulário com seu e-mail e solicita o envio.</p> <p>6) O sistema gera uma senha temporária para o usuário, alterando sua senha original e envia um e-mail ao usuário com sua nova senha.</p> <p>7) O sistema encaminha o usuário para o formulário de inserção da senha temporária.</p> <p>8) O usuário preenche o formulário com sua senha temporária e uma nova senha que será atribuída a sua conta.</p> <p>9) O sistema verifica, realiza a autenticação dos dados, registra a nova senha e encaminha o usuário para a página da agenda.</p>	
Fluxo Alternativo	
Fluxo de Execução	

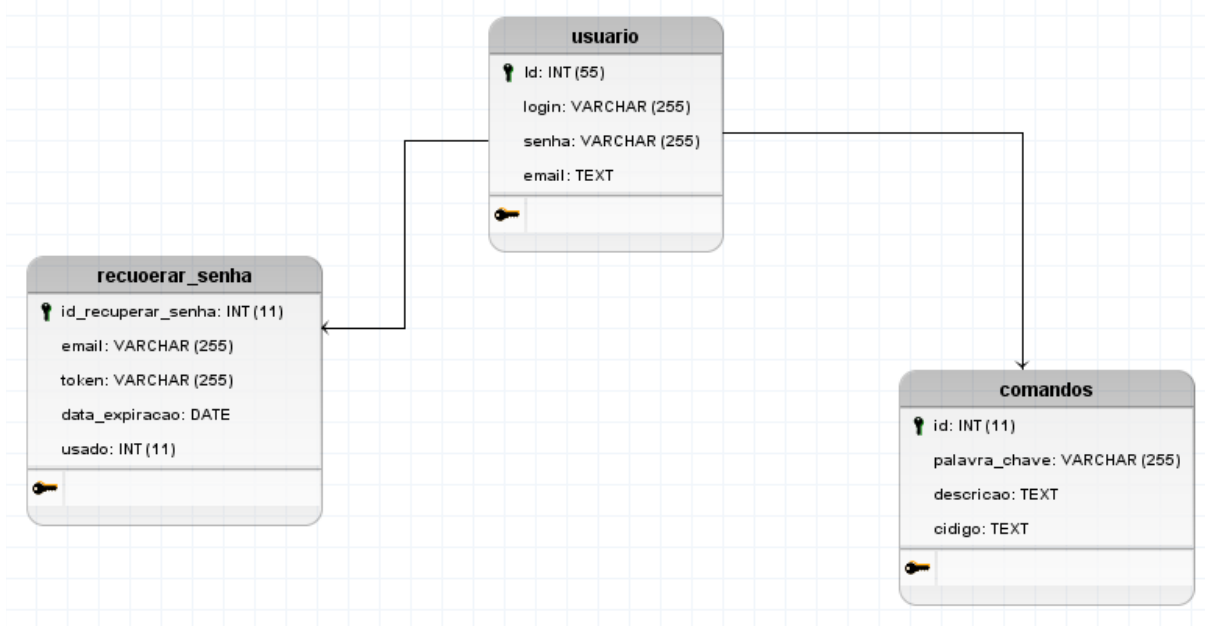
a) O sistema verifica que a senha temporária pelo usuário está incorreta e notifica o mesmo, encaminhando o usuário novamente para a página do formulário de inserção de senha temporária.

*Qualquer erro de salvamento, alteração ou consulta de dados que não foi antecipado, encaminhará o usuário para uma página de erro.

5.4 DIAGRAMA DO BANCO DE DADOS

A Figura abaixo apresenta o diagrama do banco de dados utilizado pelo sistema. Ele possui 3 tabelas, sendo elas, a tabela Usuário, que guarda as informações do usuário, como nome, senha e e-mail, a tabela Comandos que guarda todas as informações das linguagens de programação já incluídas no sistema e a tabela recuperar_senha, onde é gerado o token para recuperação da senha do usuário. Essas três tabelas são essenciais para o funcionamento do sistema, uma vez que sem as tabelas usuário e recuperar_senha, não podem ser feitas alterações ou adições de linguagens e sem a tabela comandos o sistema se torna inútil.

Figura 3: Diagrama Relacional

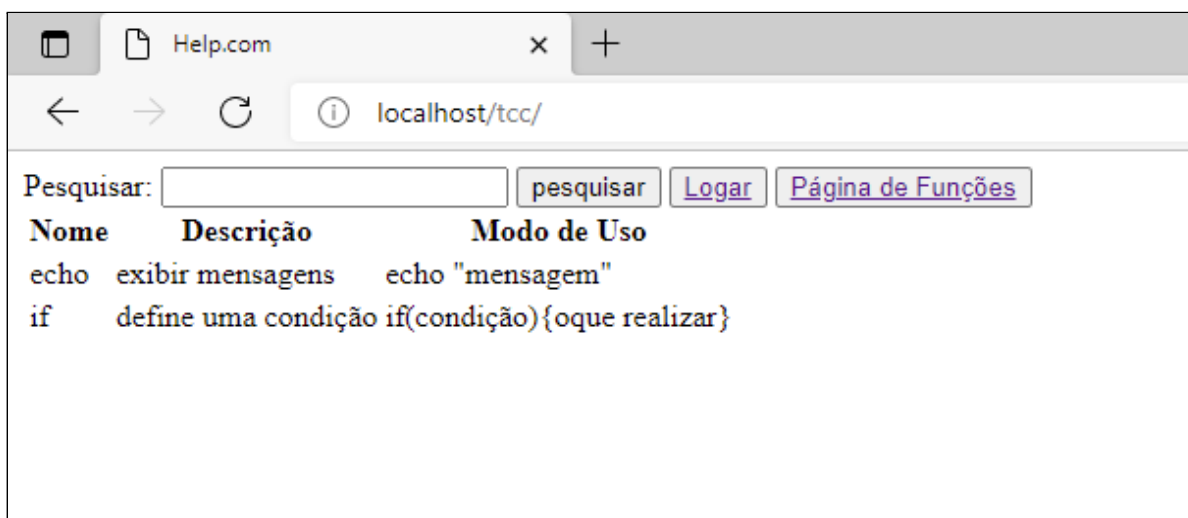


5.5 INTERFACES DO SISTEMA

Esta seção apresenta as telas de todos os requisitos funcionais essenciais do sistema. A maior parte delas foi desenvolvida com ajuda do framework materialize, com algumas pequenas especificações feitas com a linguagem JavaScript.

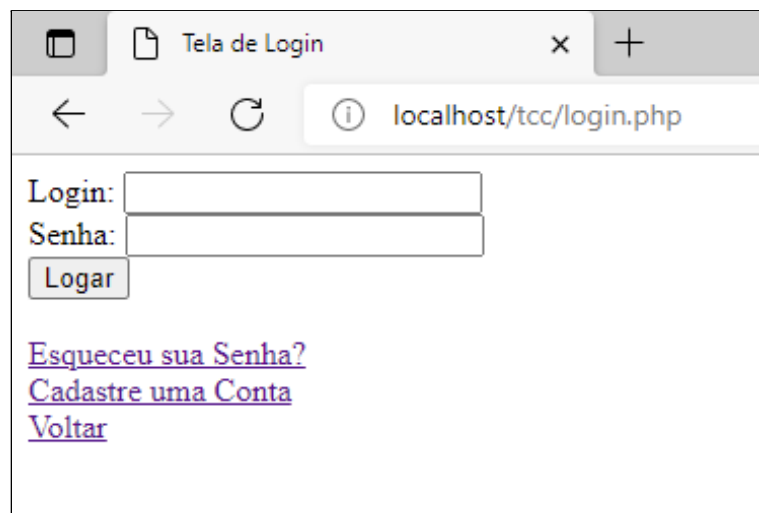
A figura 4 apresenta a tela inicial, onde o usuário poderá pesquisar e visualizar comandos e funções que desejar.

Figura 4: Tela inicial



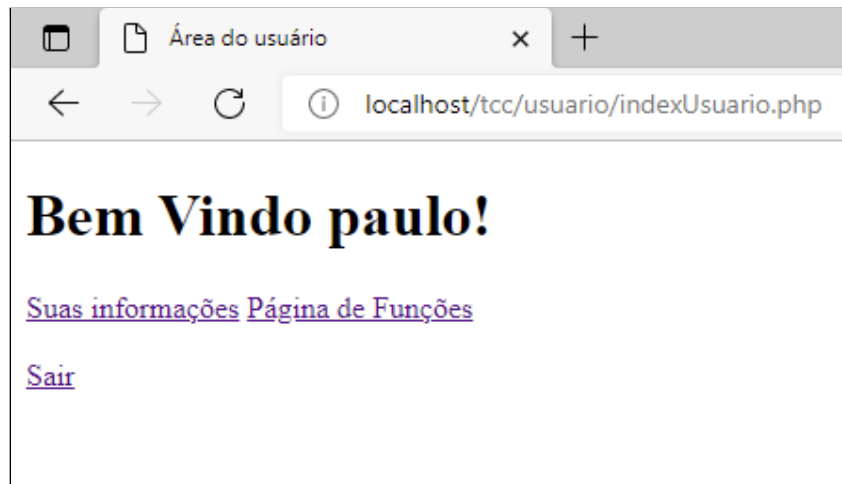
A figura 5 apresenta o formulário de login do sistema, onde o usuário ou o administrador poderá realizar login com uma conta já cadastrada no sistema.

Figura 5: Tela de login



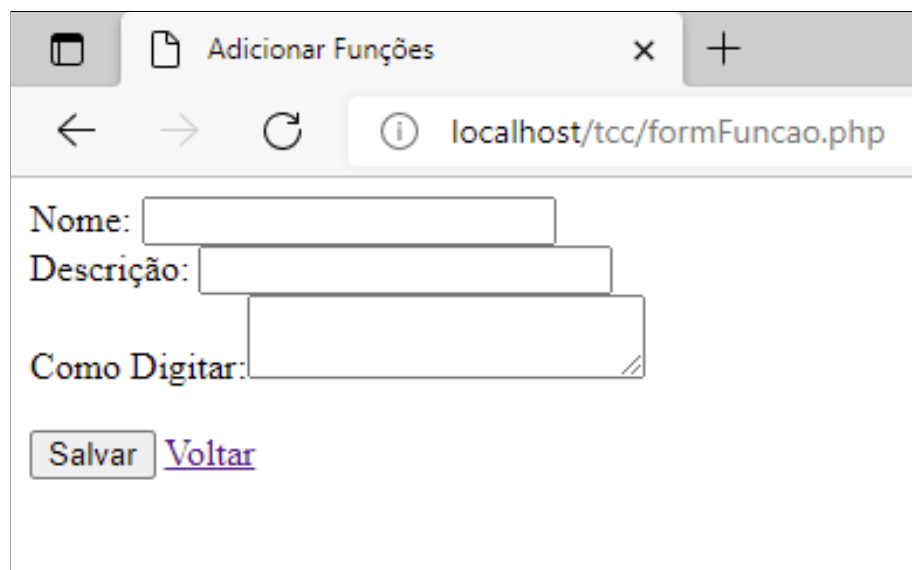
A figura 6 apresenta a tela de gerenciamento da conta do usuário, onde o usuário poderá verificar seus dados, como senha e nome ou adicionar comandos ou funções no sistema.

Figura 6: Tela de gerenciamento da conta



A figura 7 mostra o formulário de alteração ou adição de comandos, onde o usuário pode adicionar novas informações a um comando ou função, ou adicionar um comando ou função ainda não presente no sistema.

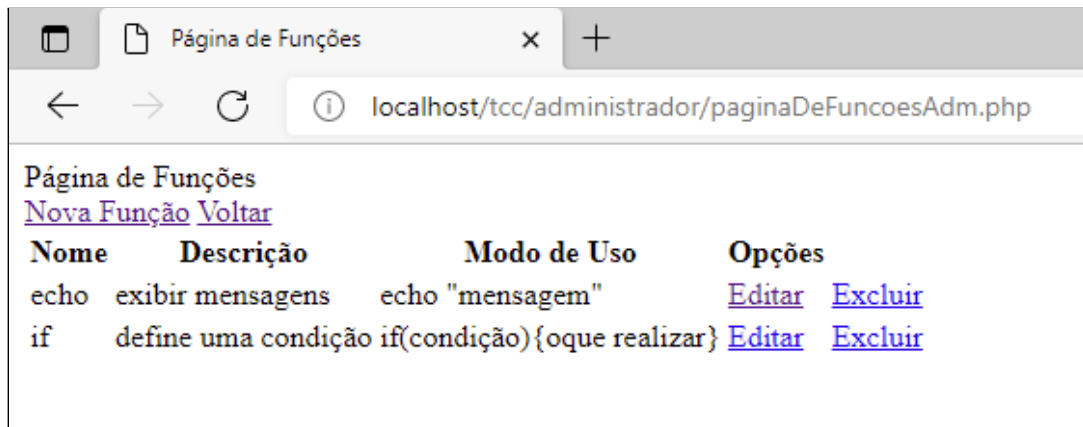
Figura 7: Tela de adição/edição de comandos



The screenshot shows a web browser window with a single tab titled "Adicionar Funções". The address bar contains "localhost/tcc/formFuncao.php". The form consists of three input fields: "Nome:", "Descrição:", and "Como Digitar:". Below the form, there are two buttons: "Salvar" and "Voltar".

A figura 8 mostra a tela do administrador, de gerenciamento de comandos ou funções, com as opções de edição de comandos e exclusão de comandos, onde o administrador pode excluir as informações de comando ou função adicionadas no sistema.

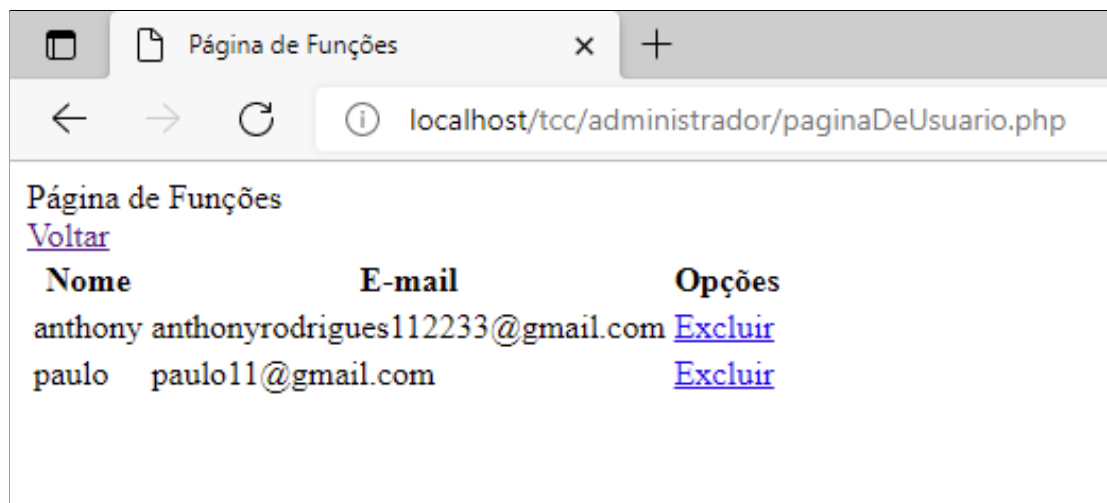
Figura 8: Tela de gerenciamento de comandos



Nome	Descrição	Modo de Uso	Opções
echo	exibir mensagens	echo "mensagem"	Editar Excluir
if	define uma condição	if(condição){oque realizar}	Editar Excluir

A figura 9 mostra a tela de gerenciamento de usuários, com a opção de exclusão de usuários, onde o administrador pode excluir as informações de comando ou função adicionadas no sistema.

Figura 9: Tela de gerenciamento de usuários



Nome	E-mail	Opções
anthony	anthonyrodrigues112233@gmail.com	Excluir
paulo	paulo11@gmail.com	Excluir

6 CONSIDERAÇÕES FINAIS

A existência de diversas linguagens de programação, bem como a vasta gama de comandos existentes em cada uma delas, faz com que a memorização das mesmas, por parte dos programadores, seja quase impossível. Fazendo-os recorrer a pesquisas na web em busca de sanar suas dúvidas com relação a “Como determinado comando funciona” ou “O que determinada função faz”.

Dito isto, o objetivo principal deste trabalho foi planejar, elaborar e desenvolver um sistema web para auxiliar programadores que venham a ter essas dúvidas. Para tanto, foram utilizadas diversas ferramentas como Materialize e JQuery além das linguagens de programação PHP e JavaScript para criação de uma biblioteca de programação onde os usuários podem sanar suas dúvidas e contribuir com o sistema adicionando novas informações sobre outras linguagens de programação ou corrigindo informações já presentes no sistema que possam vir a estar erradas ou apenas desatualizadas.

Durante o desenvolvimento do projeto foram notadas algumas dificuldades, principalmente com a linguagem JavaScript e com o Uso de Banco de Dados, os quais eu particularmente não possuía tanto conhecimento ou havia esquecido certas informações sobre. Porém com ajuda do orientador e algumas pesquisas, estas dificuldades foram superadas.

7 REFERÊNCIAS

Brainly (empresa de tecnologia educacional) pode ser encontrado em:

<<https://brainly.com.br/>>

PHP.net (site especializado em linguagem php) pode ser encontrado em:

< <https://www.php.net/> >

Wikipedia (site de informações escrito de maneira colaborativa) pode ser encontrado em:

<https://pt.wikipedia.org/wiki/Wikip%C3%A9dia:P%C3%A1gina_principal>