

Faculdade de Pará de Minas
Relatório do Projeto Aplicado

SPT

**Guilherme Augusto Fonseca Magalhães, Maycon Douglas Santos, Newton
Paula, Thiago Vasconcelos Almeida**

Orientador(a):
Reinaldo Galvão



11/09/2023



Guilherme Augusto Fonseca Magalhães, Maycon Douglas Santos,
Newton Paula, Thiago Vasconcelos Almeida

Faculdade de Pará de Minas

SPT

Relatório de Projeto Aplicado
desenvolvido para a disciplina de Projeto
Integrador

Orientador (a): Reinaldo Galvão

Pará de Minas
04/09/2023

Sumário

1. Planejamento do Projeto Aplicado	4
1.1 - Desafio	4
1.1.1 Tema Abordado	4
1.1.2 Recorte do Projeto	4
1.1.3 Benefícios e Justificativas	4
1.1.4 Análise de Contexto	4
• 1.1.4.1 Como nasceu a ideia	4
• 1.1.4.2 Problema a ser resolvido	5
• 1.1.4.3 Justificativa	5
1.2 Solução	5
1.2.1 Objetivo	5
1.2.2 Premissas e Restrições	5
1.2.3 Backlog de Produto	6
2. Área de Experimentação	7
2.1 Sprint 1	9
2.1.1 Solução	9
• Evidência do planejamento:	9
• Evidência da execução de cada requisito:	9
• Evidência dos resultados:	9
2.1.2 Lições Aprendidas	9
2.2 Sprint 2	10
2.2.1 Solução	10
• Evidência do planejamento:	10
• Evidência da execução de cada requisito:	10
• Evidência dos resultados:	10
2.2.2 Lições Aprendidas	10
2.3 Sprint 3	11
2.3.1 Solução	11
• Evidência do planejamento:	11
• Evidência da execução de cada requisito:	11
• Evidência dos resultados:	11
2.3.2 Lições Aprendidas	11
3. Considerações Finais	12
3.1 Resultados	12
3.2 Contribuições	12
3.3 Próximos passos	12

1. Planejamento do Projeto Aplicado

1.1 - Desafio

1.1.1 Tema Abordado

SPT - Sistema para pagamento de transporte.

Criação de um sistema web para compartilhamento de boletos, onde o administrador poderá encaminhar os boletos para seus usuários e um ambiente seguro, de fácil uso e acesso, buscando facilitar a tarefa e deixá-la mais intuitiva aos usuários.

1.1.2 Recorte do Projeto

De início a ideia é deixar o sistema funcional de forma local, ou seja, tendo suas funções simples de envio e recebimento de arquivos preparadas para a hospedagem na *internet*, Assim permitindo com que o sistema possa resolver a principal dificuldade do usuário, restando apenas sua hospedagem e configuração e adaptação para web, para implementações futuras.

1.1.3 Benefícios e Justificativas

Os maiores benefícios da criação do sistema são a unificação de forma de envio, possibilitando ter um ambiente específico, facilitando a busca dos usuários e o envio por parte do administrador. Aumentando a segurança de dados, pois a forma atual qualquer pessoa com acesso ao dispositivo móvel do usuário consegue coletar seus dados.

1.1.4 Análise de Contexto

- 1.1.4.1 Como nasceu a ideia

A ideia surgiu com uma situação de um integrante do grupo, que reside em outra cidade e vai para as aulas de ônibus. Esse integrante recebe os boletos por



WhatsApp, de uma maneira que é muito trabalhosa para os funcionários da empresa de transporte.

Observando a situação, surgiu a ideia de desenvolver um *website* para auxiliar no compartilhamento dos boletos, funcionando de maneira autônoma, necessitando apenas que o cliente faça *login* para baixar o boleto.

- 1.1.4.2 Problema a ser resolvido

O problema a ser resolvido é a impraticidade do método adotado pela empresa, que constitui de tarefas manuais encarregadas para um funcionário enviar cada boleto individualmente para cada cliente, que no caso demora muito tempo e é muito trabalhoso.

- 1.1.4.3 Justificativa

A solução é importante para automatizar o compartilhamento dos boletos através de um *website*, que terá benefícios para ambos os lados, tanto o cliente quanto a empresa. O uso da solução permitirá que o usuário tenha um local de fácil acesso para seus boletos mensais.

1.2 Solução

1.2.1 Objetivo

O Objetivo é criar uma ferramenta que possibilite a resolução do problema apresentado, de forma prática, estética e de baixa curva de aprendizado tanto para os usuários quanto para o administrador.

1.2.2 Premissas e Restrições

Para o bom funcionamento do projeto proposto são necessários alguns requisitos fundamentais para sua existência, performance e usabilidade. São eles:

- Bom servidor de hospedagem: Como o projeto irá trabalhar e manipular arquivos a escolha e parametrização de um bom sistema para a sua hospedagem é de suma importância, fazendo com que a análise de seus valores se torne um requisito;
- Implementação das funcionalidades: Implementar as funcionalidade de forma organizada e concisa é de grande importância, pois o sistema deverá agregar o processo e não o tornar mais difícil e complexo;
- Visual simples: O visual da ferramenta deverá ser facilitado tanto para o usuário administrador quanto para os usuários comuns;

1.2.3 Backlog de Produto

Sprint 1	<ul style="list-style-type: none">• Criação do protótipo do sistema;• Criação do Repositório;• Criação da logo;• Definição de paleta de cores;• Criar tela de Login;• Criação de telas do administrador.
Sprint 2	<ul style="list-style-type: none">• Criar Sessões;• Criação da Tela do usuário;• Criação do banco de dados (Local);• Criação de Rotas;• Ligação com o banco de dados;• Cadastro de usuários.
Sprint 3	<ul style="list-style-type: none">• Manipulação de Pastas;• Envio do arquivo;• Configurar telas(reconhecer se é o usuário ou adm);• Liberação dos pdf para o usuário;

	<ul style="list-style-type: none">• Teste da aplicação;• Otimização do projeto;
--	--

2. Área de Experimentação

O que significa esta seção?

Esta seção tem o objetivo de apresentar as evidências do planejamento dos requisitos selecionados do Backlog de Produto, além de mostrar a maneira como eles foram desenvolvidos e registrar os resultados alcançados.

É necessário expor a execução e a validação dos experimentos relacionados ao desenvolvimento da solução, ou seja, testar se você está no caminho certo ou se algo precisa ser modificado (pivotar).

Quais etapas já devem estar finalizadas no momento do preenchimento desta seção? (Pré-requisitos)

No momento do preenchimento, é esperado que você já tenha cursado a disciplina de Inovação e Design Thinking, em especial as etapas do processo de Design Thinking, além de estar se preparando para desenvolver a solução idealizada no seu Projeto Aplicado.

Você também já deve ter preenchido o primeiro capítulo deste relatório (CANVAS do Projeto Aplicado).

Como esta seção deve ser preenchida?

Esta seção é a área mais dinâmica do CANVAS do Projeto Aplicado. Nela você deverá inserir os experimentos necessários para desenvolver e validar cada Sprint. Ao final do experimento, você deverá preencher o item “**Solução**” da seguinte maneira:

- **Evidência do Planejamento:** comprove que os requisitos referentes à Sprint foram efetivamente planejados. Para isso, utilize o Trello e adicione, neste campo, uma cópia da tela da ferramenta com a Sprint planejada.
- **Evidência da Execução de cada Requisito:** para cada requisito planejado, adicione um artefato que comprove o cumprimento da etapa. Podem ser anexados, por exemplo, códigos, documentos, modelos, scripts, capturas de tela, entre outros. *Importante: o número de artefatos adicionados deve ser o mesmo que o número de requisitos planejados.*



- **Evidência da Solução:** os requisitos implementados contribuem para o alcance de um resultado geral, que deverá ser comprovado neste campo. Isso será feito por meio de capturas de tela, gráficos, modelos, textos, figuras, tabelas, testes, entre outros.

Para cada Sprint, cite no item “**Lições Aprendidas**” o que não foi validado, mas forneceu insights para ajuste da rota.

2.1 Sprint 1

2.1.1 Solução

- **Evidência do planejamento:**

Foi definido que na primeira sprint o foco seria na criação dos protótipos, levantamento de recursos e criação de diretórios que seriam exportados os códigos criados.

Com todos os recursos selecionados e a estrutura montada, iniciamos a criação das primeiras telas de administrador, focado em criar a interface das funcionalidades primordiais do projeto, que seriam o envio de arquivos, cadastro de usuário e uma busca simples pelos usuários, cadastrados pelo sistema.

Para que essas telas fiquem bonitas, padronizadas e de fácil entendimento para nossos usuários, criamos uma interface simples e uma paleta de cor baseada no tema definido pelo grupo, buscando ser de forma amigável.

- **Evidência da execução de cada requisito:**

Criação do protótipo do sistema: Criamos o protótipo pelo o figma, onde tentamos fazer o nosso projeto de acordo com a prototipação, mas adaptamos algumas coisas para melhor aproveitamento.

segue o link:

<https://www.figma.com/file/T6ghLfMXDpkPUqKOMPOKRC/Untitled?type=design&node-id=0-1&mode=design>

- Criação do Repositório: FOi criado um repositório online para que possamos realizar o controle de modificações do projeto e manter todos os integrantes do grupo com os dados e informações dos códigos;
segue o link: <https://github.com/MayconD27/SPT>

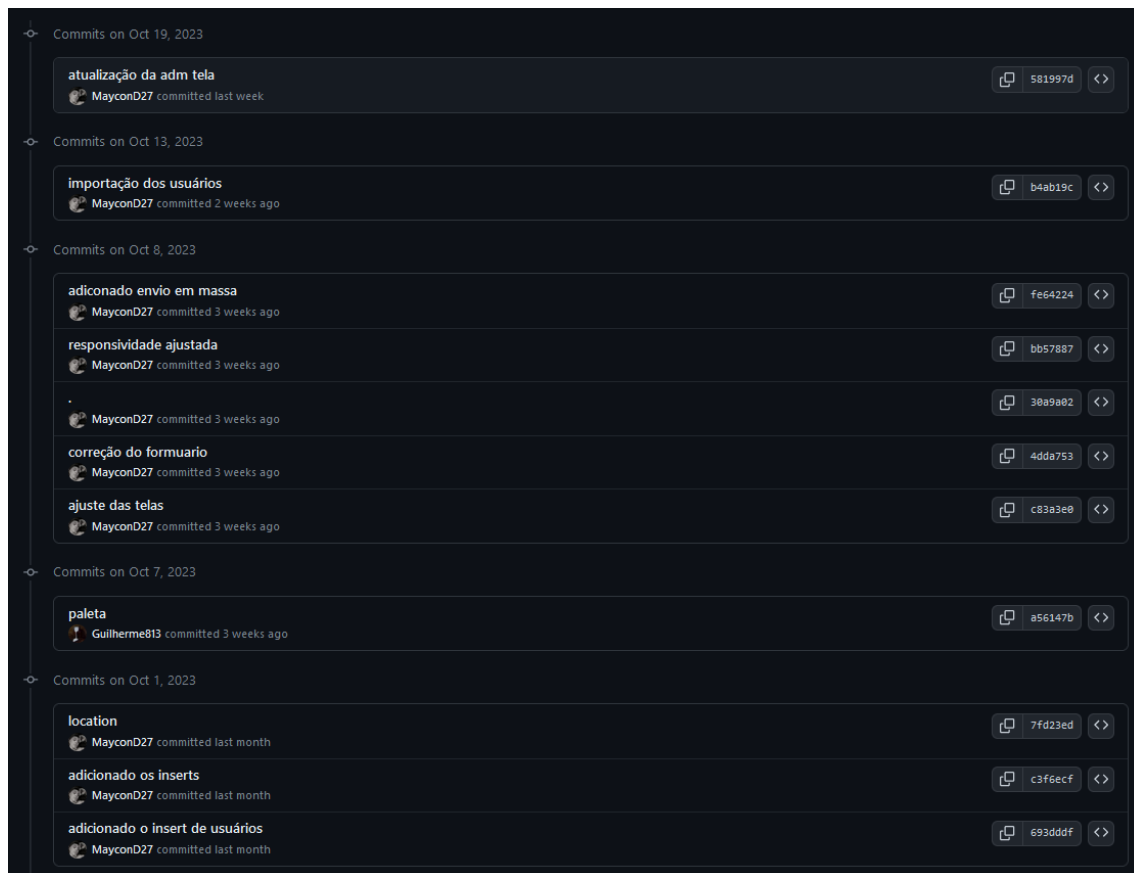
- Criação da logo;
Criamos a logo pelo [canva](#) onde buscamos uma logo simples porém que atende a proposta criada para ela.



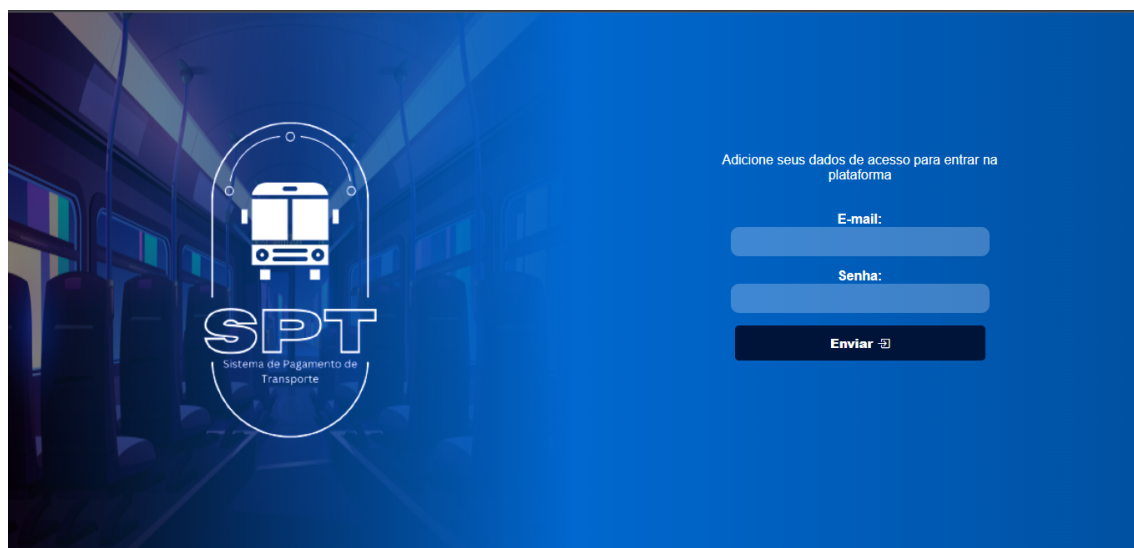


- Definição de paleta de cores: Definimos a paleta de cores utilizando um tom azul de cores.
- Criação da tela de login: Com as linguagens aprendidas criamos a tela de login e telas do administrador, dentre elas temos, css, html, php e a ferramenta online bootstrap.

- Evidência dos resultados



-



2.1.2 Lições Aprendidas

Aprendemos a utilizar uma nova linguagem de programação, o PHP, usando a lógica aprendida em sala de aula, para aplicá-la na nova linguagem. Conhecendo uma metodologia da linguagem, chamada PDO.

Aprendemos também a estrutura básica para a criação de sites, que seriam a linguagem de marcação HTML e a linguagem de estilização CSS, que foram essenciais para criação do projeto.

Também aprendemos a utilizar o software XAMPP, que auxilia no desenvolvimento quando a linguagem PHP é utilizada. Além de aprendermos de uma forma básica acessar um servidor APACHE de acesso local dentro dos computadores.

Outro aprendizado foi com o versionamento do projeto, utilizando o git e github todo o grupo pode aprender a manter o código atualizado e disponível para todos os integrantes, bem como registrar logs de uso do e versionamentos dos códigos.

Usamos o bootstrap para tornar ainda mais fácil o uso de componentes, cores, fontes e ícones no nosso projeto.

2.2 Sprint 2

2.2.1 Solução

- **Evidência do planejamento:**

Seguindo o nosso planejamento, ainda faltava fazer a tela do usuário, o banco de dados local, conectar o banco de dados com o restante do código, a sessão do usuário e o cadastro do usuário.

Como já tínhamos começado o projeto e adquirido uma básica experiência com as tecnologias envolvidas no projeto, a criação da tela foi feita rapidamente. O banco de dados local foi criado e implementado para testes, assim como a conexão do mesmo com o código antes feito. A sessão e o cadastro também foram implementados ao código, ambos completamente funcionais.

Todos os recursos propostos para implementação na segunda sprint

- Evidência da execução de cada requisito:

Tela do usuário (cliente): A tela de usuário (cliente) foi criada para o uso do usuário dentro do website, utilizando os mesmos recursos tecnológicos da criação da tela do administrador, php,css,html e bootstrap.

Banco de dados: O banco de dados local foi implementado para funções de teste na funcionalidade do website, testando campos preenchidos e a retenção das informações inseridas, para realização desta tarefa o aplicativo xampp foi extremamente importante, pois ele que nos permitiu acessar a SGBD, que nesse contexto utilizamos o PHPMyAdmin.

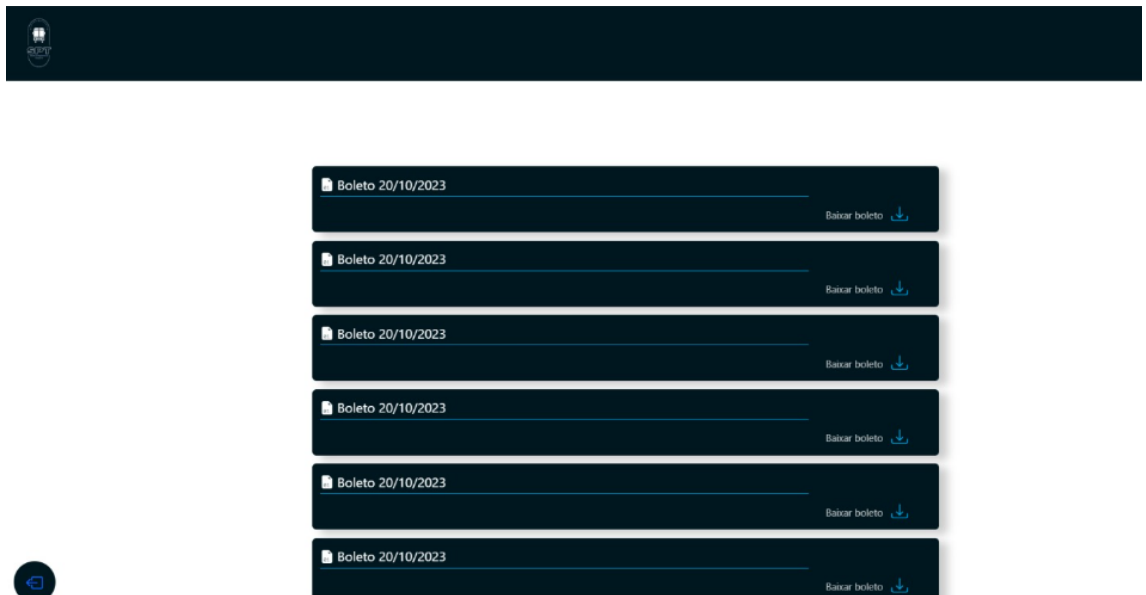
Sessões: Também foi acrescentado as sessões, que são importantes para o sistema manter os dados salvos na memória para o usuário não ter que fazer login a cada ação que realiza no site.

Cadastro de usuários: A página de cadastro de usuários será utilizada pelo administrador do sistema para cadastrar novos usuários que utilizarão o sistema.

Rotas: Foram criadas as rotas, que permitem a navegação limpa e fluída dentre as páginas do site, que determina o que será exibido na tela quando determinada URL for acessada.

Ligação de Banco de Dados: Foi feita a ligação com o banco de dados para que tenha a interação com o sistema, nesse processo utilizamos uma metodologia da linguagem PHP, essa metodologia se chama PDO.

- Evidência dos resultados:





```
session_start();
include_once "../bd.php";
    $usuarioLogado = isset($_SESSION['logado']) ? $_SESSION['logado'] : false;

    if($usuarioLogado== false){
        header('location: login.php');
        exit;
    }
    $nomeUsuario = isset($_SESSION['nome']) ? $_SESSION['nome'] : 'Sem nome';
    $id = $_SESSION['id'];

?>
```

```
<?php

define('DB_SERVER', 'localhost');
define('DB_USERNAME', 'root');
define('DB_PASSWORD', '');
define('DB_NAME', 'spt');

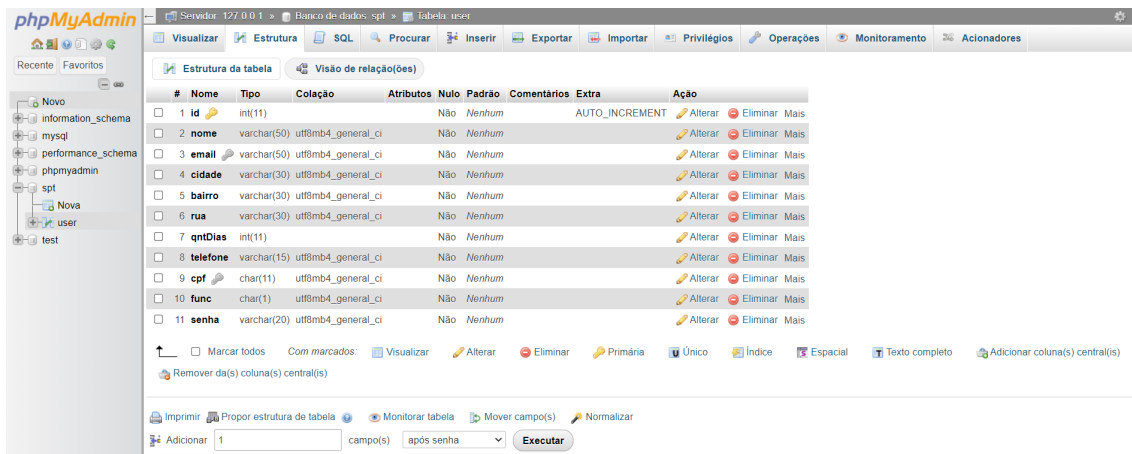
try{
    $bd = new PDO("mysql:host=" . DB_SERVER . ";dbname="
        . DB_NAME, DB_USERNAME, DB_PASSWORD, array(
            PDO::MYSQL_ATTR_INIT_COMMAND => "SET NAMES 'utf8', SESSION SQL_BIG_SELECTS=1",
            PDO::ATTR_DEFAULT_FETCH_MODE => PDO::FETCH_ASSOC,
            PDO::ATTR_PERSISTENT => true
        )
    );
    $bd->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
} catch(PDOException $e){
    die("ERROR: Não foi possível conectar." . $e->getMessage());
}
```

```
<?php
$nome_User = $_POST["nome"];
$cpf_User = $_POST["cpf"];
$email_User = $_POST["email"];
$telefone_User = $_POST["telefone"];
$qtd_User = $_POST["qtd_dias"];
$func = $_POST["func"];
$senha = $_POST["senha"];
$bairro = $_POST["bairro"];
$cidade = $_POST["cidade"];
$rua = $_POST["rua"];

include_once "../bd.php";

$sql = "INSERT INTO user VALUE (NULL, '$nome_User', '$email_User', '$cidade', '$bairro', '$rua', $qtd_User, '$telefone_User', '$cpf_User', '$func'";
$resultado = $bd->prepare($sql);
$registro = $resultado->execute();
header('location: cadastro_usuario.php');

?>
```



The screenshot shows the phpMyAdmin interface for a MySQL database named 'spt'. The 'user' table is selected, and its structure is displayed. The table has 11 columns: id, nome, email, cidade, bairro, rua, qntDias, telefone, cpf, func, and senha. The 'id' column is the primary key and is auto-incrementing. The 'nome', 'email', 'cidade', 'bairro', 'rua', 'telefone', 'cpf', 'func', and 'senha' columns are all of type VARCHAR or CHAR and are not nullable.

#	Nome	Tipo	Colação	Atributos	Nulo	Padrão	Comentários	Extra	Ação
1	id	int(11)			Não	Nenhum		AUTO_INCREMENT	Alterar Eliminar Mais
2	nome	varchar(50)	utf8mb4_general_ci		Não	Nenhum			Alterar Eliminar Mais
3	email	varchar(50)	utf8mb4_general_ci		Não	Nenhum			Alterar Eliminar Mais
4	cidade	varchar(30)	utf8mb4_general_ci		Não	Nenhum			Alterar Eliminar Mais
5	bairro	varchar(30)	utf8mb4_general_ci		Não	Nenhum			Alterar Eliminar Mais
6	rua	varchar(30)	utf8mb4_general_ci		Não	Nenhum			Alterar Eliminar Mais
7	qntDias	int(11)			Não	Nenhum			Alterar Eliminar Mais
8	telefone	varchar(15)	utf8mb4_general_ci		Não	Nenhum			Alterar Eliminar Mais
9	cpf	char(11)	utf8mb4_general_ci		Não	Nenhum			Alterar Eliminar Mais
10	func	char(1)	utf8mb4_general_ci		Não	Nenhum			Alterar Eliminar Mais
11	senha	varchar(20)	utf8mb4_general_ci		Não	Nenhum			Alterar Eliminar Mais

2.2.2 Lições Aprendidas

As lições aprendidas incluem a continuidade no uso e aprendizado de tecnologias front-end como PHP, CSS, HTML e Bootstrap para criar uma interface amigável. A criação de um banco de dados local foi fundamental tanto para aprendermos a lidar com a criação de queries quanto na prática dos comandos de banco de dados aprendidos em sala de aula. Aprender o uso de sessões foi algo muito prático para entendermos como funciona o sistema de login e logout de um sistema mantendo dados na memória, melhorando a experiência do usuário. Criar as páginas de cadastro de usuários foi importante para aprendermos os conceitos de GET e POST, que são de suma importância quando abordamos o tema formulários. A ligação com o banco de dados usando PDO foi o conceito mais complexo, pois foi necessário entender o básico de como o template fornecido interage com as queries e como podemos alterar as informações do template para as informações de nosso banco local. No geral, as lições enfatizam boas práticas de desenvolvimento web, usabilidade e interação eficaz com dados.