

# SAW – Trabalho Prático

SEGURANÇA DAS APLICAÇÕES WEB

ANDRÉ PINTO (8200613) / SERGIO FÉLIX (8200615)

## Índice

Introdução.....	2
Requisitos.....	2
Arquitetura do Projeto.....	3
Método de Funcionamento.....	3
Implementação e Segurança.....	5
Roles e Áreas Restritas.....	5
Sanitização de Dados.....	6
Validação de Dados.....	7
Register.....	8
Atempt Login.....	9
Remember me.....	10
Encriptação Cookies.....	10
Recover Password.....	11
Error Handling.....	13
Logs / Erros.....	14
Bloqueio de Utilizadores.....	15
Upload de Imagens.....	16
Conclusão.....	17
Bibliografia.....	18
Apresentações da Disciplina SAW.....	18
Documentação PHP.....	18
Pesquisa e Aprendizagem.....	18



## Introdução

No âmbito da disciplina de Segurança das Aplicações WEB do 2º ano, 1º semestre do Curso Técnico Superior Profissional em Desenvolvimento para a Web e Dispositivos Moveis, foi pedido o desenvolvimento de uma aplicação Web em PHP, que simule uma aplicação de venda de produtos usados. Neste, deve ser utilizado todo o conhecimento adquirido nas aulas, implementando soluções de segurança aprendidas e que contemplasse todas as funcionalidades pedidas no enunciado.

## Requisitos

Esta aplicação Web foi exclusivamente desenvolvida em código PHP 8. Para tal, será necessário ter em conta alguns requisitos para conseguir executar a aplicação. As vulnerabilidades não estão apenas no código implementado, algumas delas são provocadas por configurações defeituosas nos servidores que abrem portas para invasões. Para tal segue todos os requisitos necessários para correr a aplicação de forma mais segura.

Em primeiro lugar é necessário disponibilizar um serviço Apache para correr o código PHP 8. Para tal foi utilizado o programa XAMP, em que é possível ligar um serviço desta natureza. Para além disso, foram aplicadas algumas configurações para aumentar segurança e a confidencialidade dos dados, nomeadamente:

- Desativação da lista de diretórios.
- Desativação de módulos e serviços PHP desnecessários.
- Desativação do serviço Apache server-info e server status.
- Desativação da diretiva ServerSignature.
- Desativação do rastreamento HTTP Request.
- Redução do tempo de timeout para prevenir ataques DoS.
- Ligação HTTPS – para permitir a encriptação dos dados de ponto a ponto (User - Servidor).

Em relação á persistência de dados, foi utilizado um banco de dados MySQL. Para ativar este serviço, igualmente ao servidor Apache, foi utilizado o programa XAMP, que possibilita a criação deste serviço. Foram aplicadas algumas configurações para aumentar segurança e a confidencialidade dos dados, nomeadamente:

- Alteração da password da conta ROOT
- Criação de um utilizador administrador, com todas as permissões apenas para a base de dados a utilizar.
- Criação uma conta apenas com permissões de leitura e escrita na base de dados a utilizar para configurar na aplicação.

Assim, são estes os requisitos básicos e alterações necessárias para aumentar a segurança da nossa aplicação WEB.

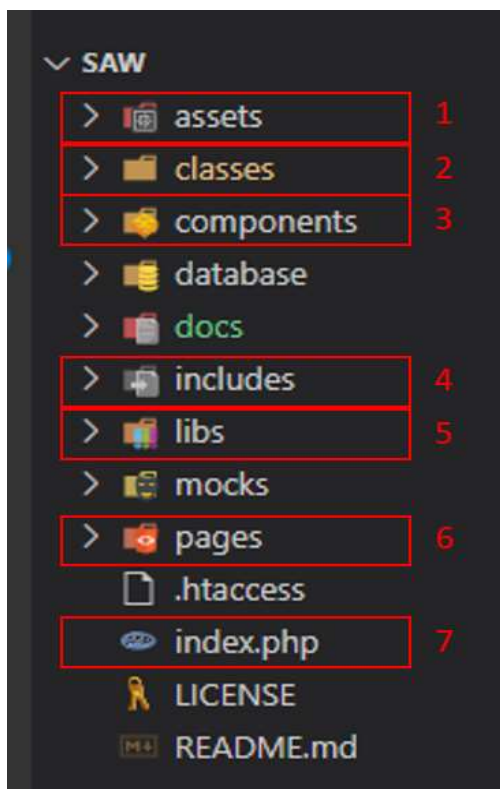
## Arquitetura do Projeto

Este trabalho, não seria um projeto de grandes dimensões, mas por nossa opção decidimos criar uma estrutura do projeto que fosse capaz de organizar todos os módulos necessários de uma forma intuitiva e organizada.

A estrutura criada é sólida e escalável. Temos os módulos devidamente separados a funcionar de forma independente. Assim, permite-nos uma maior reutilização de código e uma melhor legibilidade do projeto.

## Método de Funcionamento

### Pastas



1. Pasta para armazenar, scripts, estilos e as imagens incluindo os uploads.
2. Classes responsáveis pela logica da aplicação e inclui validação dos dados
3. Componentes de layout. Navbar , Footer...
4. Classes auxiliares, normalmente funções reutilizáveis.
5. Bibliotecas externas
6. Páginas a serem carregadas. Inclui sanitização dos dados.
7. Ficheiro inicial da aplicação

## Index.php

O projeto inicia-se no ficheiro “index.php”. Este, apenas é responsável por iniciar a aplicação. Cria um objeto do tipo “Application”, em que a suas principais funções é explodir o link URL para possibilitar uma navegação por rotas (1). Também é possível definir alguns parâmetros globais para posteriormente serem utilizados em outros locais (1). Por fim importa o ficheiro que contem o layout (2)

```
classes > application.php
1  <?php
2  class Application
3  {
4      private function __clone()
5      {
6      }
7
8      public function __construct()
9      {
10         $urlParts = array();
11         if (isset($_GET['url'])) {
12             $urlParts = explode('/', $_GET['url']);
13         }
14         define('APP_URL_PARTS', $urlParts);
15         define('HOME_URL_PREFIX', '/SAW');
16     }
17
18     public function startApp()
19     {
20         require_once('./components/Layout/layout.php');
21     }
22 }
23
```

## layout.php

Em relação ao ficheiro do “layout.php”, é responsável por importar a estrutura da aplicação. É visível que importa o componente da NavBar e o Footer. O mais importante é que este ficheiro importa o controlador das páginas. O que possibilita as rotas da aplicação.

Consoante o URL, a aplicação carrega páginas diferentes, permitindo a navegação sem ter de repetir o NavBar e o Footer.

```
components > Layout > layout.php
1  <!DOCTYPE html>
2  <html lang="en-US">
3
4  <head>
5      <title>SAW</title>
6      <meta charset="utf-8" />
7      <meta name="author" content="André & Sérgio" />
8      <meta name="description" content="This is an evaluation pr">
9      <meta http-equiv="X-UA-Compatible" content="IE=edge" />
10     <meta name="viewport" content="width=device-width, initial
11     <link rel="icon" type="image/svg" href="<?php echo HOME_UR
12     <link rel="stylesheet" type="text/css" href="<?php echo HO
13     <link rel="stylesheet" type="text/css" href="<?php echo HO
14     <link rel="stylesheet" type="text/css" href="<?php echo HO
15 </head>
16
17 <body>
18     <?php
19     include_once('./components/Navbar/navbar.php');
20     include_once('./includes/page.php');
21     include_once('./components/Footer/footer.php');
22     >
23     <script src="<?php echo HOME_URL_PREFIX;>assets/scripts/
24     <script src="<?php echo HOME_URL_PREFIX;>assets/scripts/
25 </body>
26
27 </html>
```

```
includes > page.php
1  <?php
2
3  $page = isset(APP_URL_PARTS[0]) ? APP_URL_PARTS[0] : null;
4
5  switch ($page) {
6      case 'signup':
7          include_once('./pages/SignUp/signup.php');
8          break;
9      case 'signin':
10         include_once('./pages/SignIn/signin.php');
11         break;
12     case 'signout':
13         include_once('./pages/SignOut/signout.php');
14         break;
15     case 'account':
16         include_once('./pages/Account/account.php');
17         break;
18     case 'myproducts':
19         include_once('./pages/MyProducts/myproducts.php');
20         break;
21     case 'sell':
22         include_once('./pages/Sell/sell.php');
23         break;
24     case 'products':
25         include_once('./pages/Products/products.php');
26         break;
27 }
```

## Implementação e Segurança

A implementação de sistemas de segurança foi o principal objetivo deste trabalho. Para tal, implementamos vários conceitos aprendidos durante as aulas.

### Roles e Áreas Restritas

A aplicação conta com a implementação de restrições de acesso e roles. Foram definidos 3 grandes grupos de autenticação distintos, em que cada um terá diferentes permissões de acesso às páginas. Os grupos de acesso são os seguintes:

- **Utilizadores não autenticados** - Todos os utilizadores que não possuem autenticação, poderão apenas navegar pelos artigos listados pelos vendedores. No entanto não serão mostrados todos os dados do vendedor, visto que é um utilizador não autenticado, não será mostrado o contacto do vendedor.
- **Utilizadores autenticados** - Todos os utilizadores que possuem autenticação, poderão navegar pelos artigos sem nenhuma restrição. Para além disso têm acesso ao painel de administração, onde podem criar artigo, alterar os seus artigos e visualizar e alterar o seu perfil. Por questões de segurança, não é possível alterar o email.
- **Utilizadores administradores** – Um utilizador Administrador, herda todas as características de um utilizador autenticado. No entanto tem a possibilidade, através do painel de administração, visualizar todos os utilizadores, podendo bloquear ou desbloqueá-los. Como funcionalidade extra, para os administradores também é possível aceder à página de administração de log, onde podem monitorizar o funcionamento da aplicação.

## Sanitização de Dados

O primeiro ponto a ser implementado foi a sanitização dos dados. Esta funcionalidade visa filtrar todos os inputs do utilizador para evitar ataques como SQL Injection entre outros. Após o utilizador inserir os dados, a primeira camada é imediatamente a sanitização e está colocada em todos os locais onde existe inputs do utilizador.

Para sua implementação, passamos todos os valores para o array `$data`. De seguida, no array `$args`, definimos o tipo de sanitização que pretendemos aplicar. Por último realizamos `filter_var_array` para executar as sanitizações.

Este procedimento é incluído em todos os inputs do utilizador.

```
$categories = $category->getcategories();
if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['submit'])) {
    $data = array(
        'name' => $_POST['name'],
        'category' => $_POST['category'],
        'images' => $_FILES['images'],
        'price' => $_POST['price'],
        'description' => $_POST['description'],
        'user' => $_SESSION['id']
    );
    $args = array(
        'name' => FILTER_SANITIZE_STRING,
        'category' => FILTER_SANITIZE_NUMBER_INT,
        'price' => FILTER_SANITIZE_NUMBER_INT,
        'description' => FILTER_SANITIZE_STRING,
        'user' => FILTER_SANITIZE_NUMBER_INT
    );
    $cleanData = filter_var_array($data, $args);
    $cleanData += ['images' => $data['images']];
    if (!$cleanData) {
        header('location: ' . HOME_URL_PREFIX . '/sell?error');
    }

    $sell = new Sell($cleanData);
    $sell->sellProduct();
    header('location: ' . HOME_URL_PREFIX . '/myproducts');
}
```

## Validação de Dados

Com a validação de dados, permite-nos restringir as características dos valores que o utilizador pode inserir. Com isto podemos definir limites ou obrigatoriedades para os valores inseridos como a password ter obrigatoriamente 8 dígitos e símbolos.

Para sua implementação, incluímos para cada parâmetro as verificações necessárias para a sua validação. Caso esta falhe, será utilizado um sistema de ErrorHandling para mostrar ao utilizador os parâmetros inválidos.

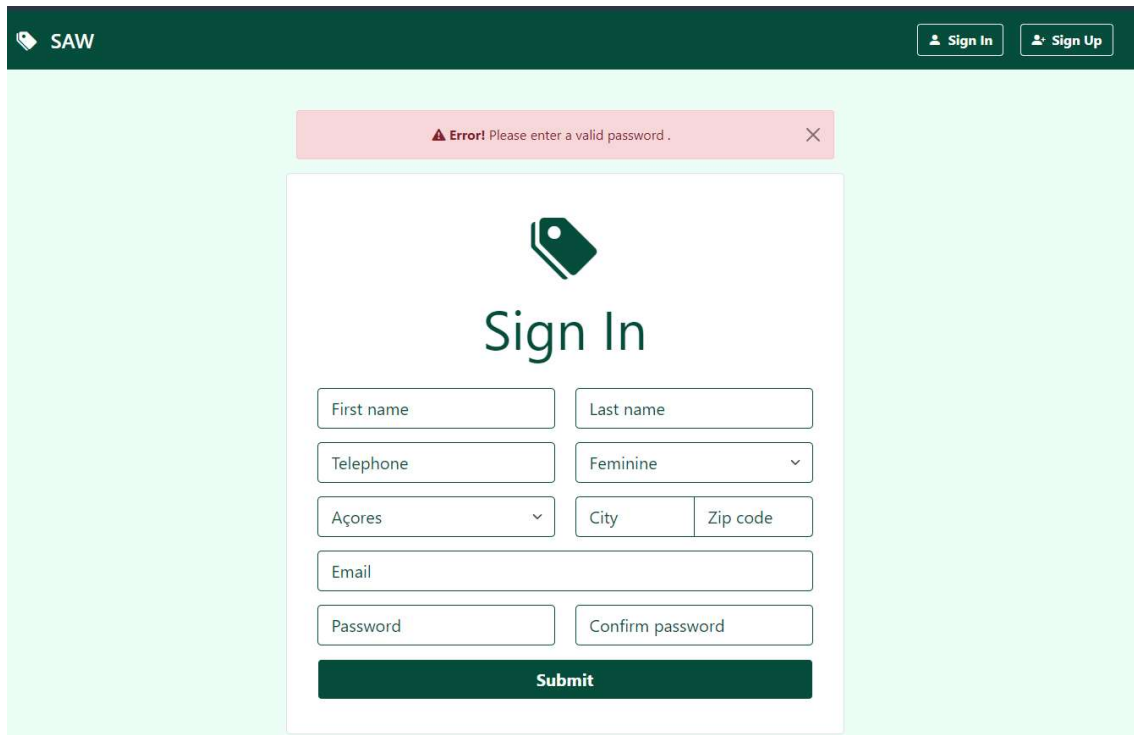
Esta implementação é colocada para todos os dados inseridos pelo utilizador.

```
if ($this->isEmpty()) {
    header('location: ' . HOME_URL_PREFIX . '/signup?error=inputs');
    exit();
}
if ($this->firstNameInvalid()) {
    header('location: ' . HOME_URL_PREFIX . '/signup?error=firstname');
    exit();
}
if ($this->lastNameInvalid()) { ...
}
if ($this->telephoneInvalid()) { ...
}
if ($this->genderInvalid()) { ...
}
if ($this->stateInvalid()) { ...
}
if ($this->cityInvalid()) { ...
}
if ($this->zipCodeInvalid()) { ...
}
if ($this->emailInvalid()) { ...
}
if ($this->passwordInvalid()) { ...
}
if ($this->passwordMismatched()) { ...
}
if ($this->userTaken()) { ...
}
private function lastNameInvalid()
{ ...
}
private function telephoneInvalid()
{ ...
}
private function genderInvalid()
{ ...
}
private function stateInvalid()
{ ...
}
private function cityInvalid()
{ ...
}
private function zipCodeInvalid()
{ ...
}
private function emailInvalid()
{ ...
}
private function passwordInvalid()
{
    if (!preg_match('/^(?=.*[a-z])(?=.*[A-Z])(?=.*\d)(?=.*[@$!%*?&])[a-zA-Z\d@$!%*?&]{8,}/', $this->password)) {
        $result = true;
    } else {
        $result = false;
    }
    return $result;
}
private function passwordMismatched()
{ ...
}
```



## Register

O registo de utilizadores é umas das funcionalidades pedidas do enunciado, que do qual é necessário ter em atenção alguns critérios para manter a segurança da aplicação.



SAW

Sign In Sign Up

Error! Please enter a valid password .

Sign In

First name Last name

Telephone Feminine

Açores City Zip code

Email

Password Confirm password

Submit

Este formulário é responsável por recolher os dados inseridos pelo utilizador e criar um novo registo na base de dados. Mas, antes disso, são aplicados vários passos de segurança. Nomeadamente, sanitização para prevenir ataques de SQL Injection e validação para restringir possíveis dados que o utilizador coloque que não possam ser aceites.

Após passar por estas validações o utilizador já pode ser criado. No entanto, temos de ter em conta uma característica importante. Temos de garantir a confidencialidade da password do utilizador. Devido á sua enorme importância é necessário garantir a confidencialidade, mesmo em caso de vazamento de dados. Assim é necessário armazenar de forma encriptada este valor na base de dados. Para tal, em vez da password guardamos um HASH dela própria. Utilizamos uma função PHP “password\_hash” com o algoritmo “bcrypt” que nos fornece um hash de 128-bit, perfeitamente capaz para a nossa necessidade.

Assim garantimos que o processo de descriptação é impossível. Alguma comparação futura terá de ser através da password original que apenas o utilizador conhece.

## Atempt Login

Os ataques de “BrutForce” são uma realidade, para prevenir, implementar uma camada de segurança á nossa aplicação para conseguir mitigar este problema. Estes ataques provem de inúmeras tentativas de login para conseguir adivinhar a password do utilizador. Para tal, definimos que a password tem de cumprir um conjunto de requisitos mínimos.

A password tem de conter letras maiúsculas e minúsculas, números, caracteres especiais e pelo menos 8 caracteres. Implementação feita através da camada de validação de dados que realizamos em todas as entradas de dados do utilizador.

```
private function passwordInvalid()
{
    if (!preg_match('/^(?=.*[a-z])(?=.*[A-Z])(?=.*\d)(?=.*[@$!%*?&])[a-zA-Z\d@$!%*?&]{8,}/', $this->password)) {
        $result = true;
    } else {
        $result = false;
    }
    return $result;
}
```

Para além de obrigar o utilizador a utilizar uma password mais segura, adicionamos um sistema de “AtemptLogin”. Sempre que um utilizador realiza uma tentativa de Login falhada, é registado na base de dados. Caso exista 3 tentativas falhadas nos últimos 1 horas, a conta é bloqueada. Apenas é possível realizar login na hora seguinte.

Este procedimento visa o bloqueio da conta caso o utilizador exceda o número de tentativas máximo.

✓ A mostrar registos de 0 - 2 (3 total, A consulta demorou 0,0005 segundos.)

SELECT \* FROM `attempt`

Perfil

[ Editar em linha ]

[ Edita ]

[ Explicar SQL ]

[ Criar código PHP ]

[ Atualizar ]

☐ Mostrar tudo

Número de registos: 25

Filtrar registos:

+ Opções

←

→

☐

 Edita

 Copiar

 Apagar

62

8200615@estg.ipp.pt

2022-01-09 23:48:06

☐

 Edita

 Copiar

 Apagar

63

8200615@estg.ipp.pt

2022-01-09 23:48:07

☐

 Edita

 Copiar

 Apagar

64

8200615@estg.ipp.pt

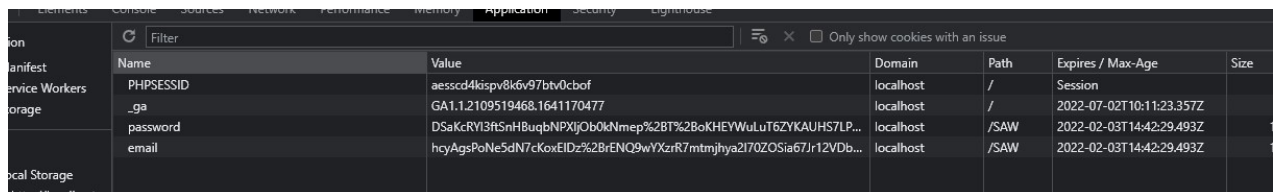
2022-01-09 23:48:08

```
$stmt = $this->connect()->prepare('SELECT COUNT(*) AS attempts FROM attempt WHERE date >= DATE_SUB(NOW(),INTERVAL 1 HOUR) AND email = ?;');
if (!$stmt->execute(array($this->email))) {
    $stmt = null;
    header('location:'. HOME_URL_PREFIX . '/signin?error=stmtfailed'); "stmtfailed": Unknown word.
    exit();
}
```

## Remember me

No meio de tanta segurança, a usabilidade pode ficar um pouco afetada, para isso temos de adicionar funcionalidade que melhorem a usabilidade do utilizador. Para tal, colocamos a propriedade “Remember Me” na magina de login. Assim o utilizar não necessita de escrever sempre os dados de log in por estes estão guardados em cookies. Caso estes dados não estivessem encriptados, estaríamos a cometer uma enorme falha de segurança pois estes podem ser facilmente acedidos.

Para tentar garantir uma maior proteção para o utilizador utilizamos algoritmos mais complexos e modernos de encriptação. Estes dados são guardados em cookies após passarem pelo um processo de encriptação e é agendado um prazo para a cookie expirar.



Name	Value	Domain	Path	Expires / Max-Age	Size
PHPSESSID	aesscd4kispv8k6v97btv0cbf	localhost	/	Session	
_ga	GA1.1.2109519468.1641170477	localhost	/	2022-07-02T10:11:23.357Z	
password	DSaKcRYI3R5nH8uqbNPXjJO60kNmep%2BT%2BoKHEYWuLuT6ZYKAUHS7LP...	localhost	/SAW	2022-02-03T14:42:29.493Z	
email	hcyAgsPoNe5dN7cKoxEIDz%2BrENQ9wYXzrR7mtmjhya2I70ZOSia67Jr12VDb...	localhost	/SAW	2022-02-03T14:42:29.493Z	

## Encriptação Cookies

Inicialmente utilizamos a função PHP “openssl\_encrypt” com uma chave (\$key), cifra “AES-128-CBC” com o algoritmo “sha256”. Pós este passo, geramos um hash das propriedades de encriptação do primeiro passo e voltamos a realizar uma segunda camada de encriptação através da função PHP “base64\_encode” em que a chave de encriptação é o hash gerado anteriormente.

Aplicando este 2 nível de encriptação, aumentamos a segurança dos dados que estão bastante expostos.

Para a desencriptação é aplicado exatamente o processo oposto.

```
includes > crypto.php
1  <?php
2  function encryptData($data)
3  {
4      $key = 'YQUaTz9-4W4xyurv';
5      $cipher = 'AES-128-CBC';
6      $algorithm = 'sha256';
7      1 $ivLength = openssl_cipher_iv_length($cipher);
8      $iv = openssl_random_pseudo_bytes($ivLength);
9      $options = OPENSSL_RAW_DATA;
10     $cipherTextRaw = openssl_encrypt($data, $cipher, $key, $options, $iv);
11     $as_binary = true;
12     2 $hmac = hash_hmac($algorithm, $cipherTextRaw, $key, $as_binary);
13     3 $cipherText = base64_encode($iv . $hmac . $cipherTextRaw);
14
15     return $cipherText;
16 }
17
18 function decryptData($data)
19 {
```

## Recover Password

Possibilitar ao utilizador a recuperação da password é uma funcionalidade muito importante, que da qual, necessita especial atenção para a sua implementação. É necessário garantir que todo o processo seja fácil e intuitivo sem esquecer que manter a segurança acima de tudo para evitar roubos ou acesso a informação privada.

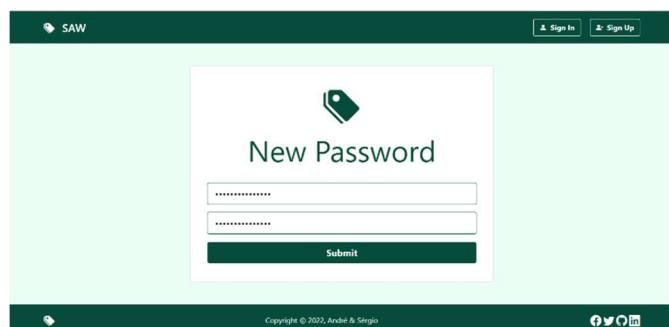
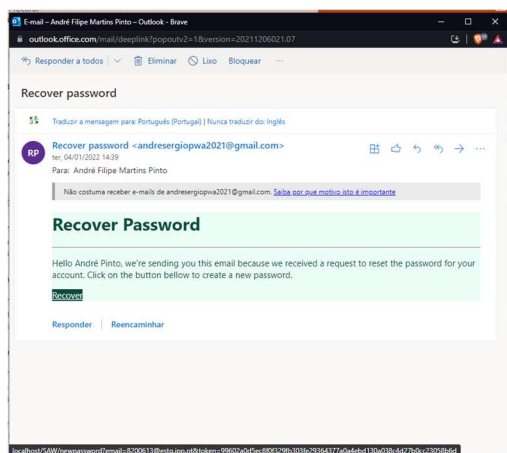
Inicialmente, o utilizador insere o email e realiza um pedido de alteração de password.



The screenshot shows the 'Forgot Password' form on the SAW website. The form is centered on a light green background. It features a dark green header with the SAW logo and 'Sign In' / 'Sign Up' buttons. The form itself has a white background and contains a text input field with the email '8200613@estg.ipp.pt' and a 'Submit' button. The footer of the page includes a copyright notice 'Copyright © 2022, André & Sérgio' and social media icons.

Posteriormente, a aplicação irá tomar uma série de paços para responder a este pedido:

- Realiza a sanitização e validação do email
- Verifica a existência do email
- Elimina pedidos de recuperação anteriores que possam existir na tabela.
- Gera um token aleatório de 32 bytes para que posteriormente este seja enviado para o utilizador e seja armazenado na base de dados passando pelo um processo de HASH através da função PHP “password\_hash” com o algoritmo “bcrypt” que nos fornece um hash de 128-bit, perfeitamente capaz para a nossa necessidade.
- Envia um email ao utilizador com os próximos passos a seguir. Assim garantimos que esta alteração apenas será feita pelo proprietário.

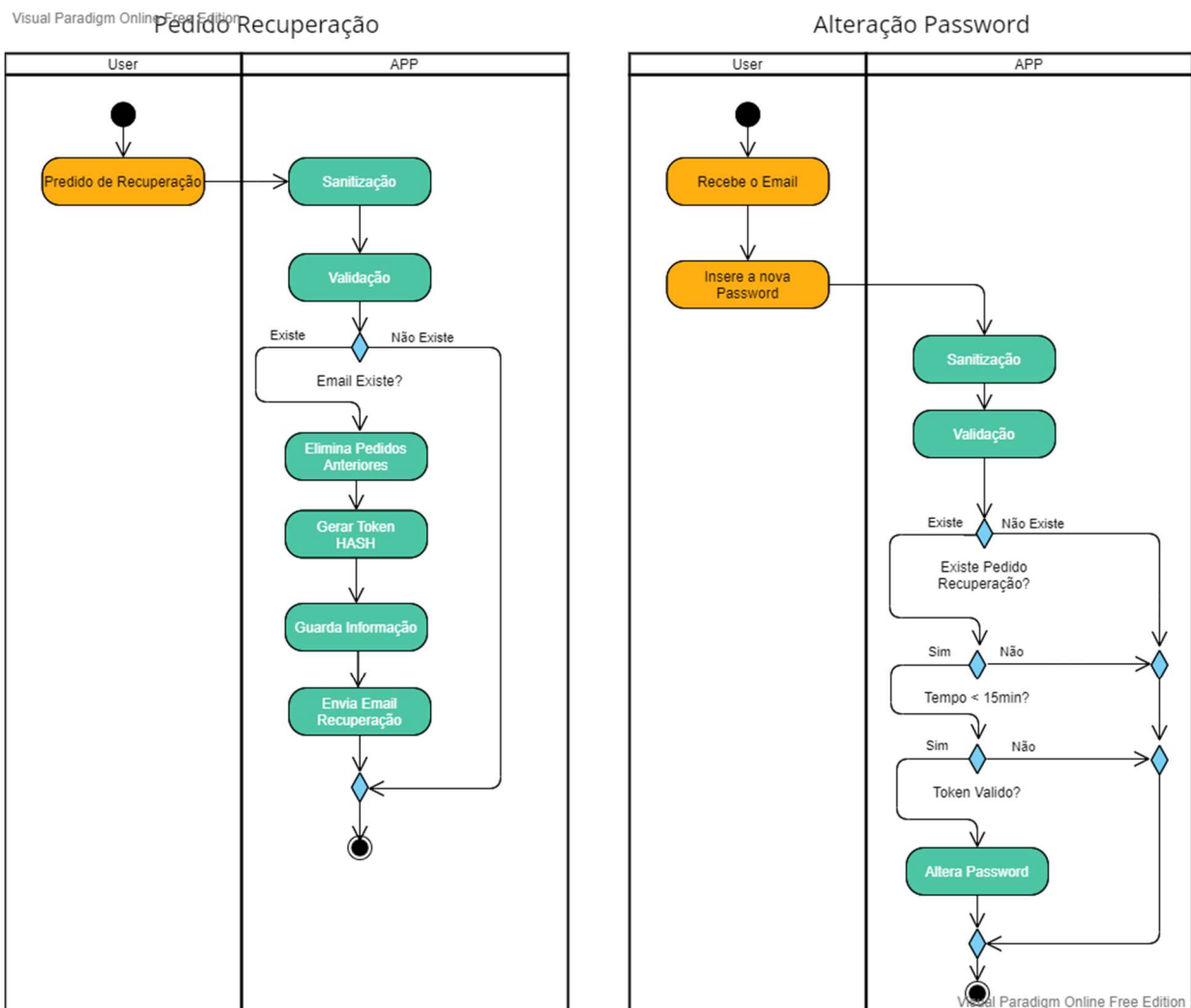


The screenshot shows the 'New Password' form on the SAW website. The form is centered on a light green background. It features a dark green header with the SAW logo and 'Sign In' / 'Sign Up' buttons. The form itself has a white background and contains two text input fields for password and confirmation, and a 'Submit' button. The footer of the page includes a copyright notice 'Copyright © 2022, André & Sérgio' and social media icons.

Seguidamente o utilizador abre o link indicado no email e será redirecionado para a página de alteração da password e o utilizador terá de colocar a nova password e efetivar a alteração. Caso o token presente no link não seja válido ou tenha excedido os 15 min para realizar a recuperação o utilizador é automaticamente redirecionado para a página de login e não lhe é permitido realizar a alteração.

Todos os dados também são sanitizados e posteriormente validados e por fim é efetivada a alteração da password.

### Diagrama de Atividades de todo o processo de recuperação.

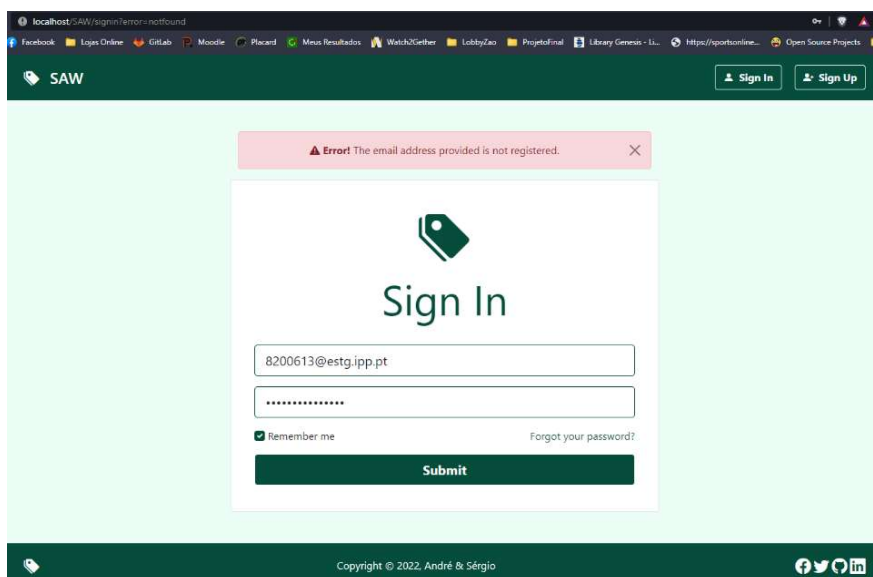


## Error Handling

Esta é uma funcionalidade básica, utilizada para conseguir mostrar aos utilizadores erros que possam estar a acontecer durante a utilização da aplicação. Para tal, criamos numa classe responsável pela tradução de códigos de erros internos para mensagens visíveis para o utilizador. Assim quando acontecer algum erro, redirecionamos o utilizador para a mesma página com o parâmetro no URL “error” com o código do erro. Posteriormente é apresentado a mensagem de erro ao utilizador.

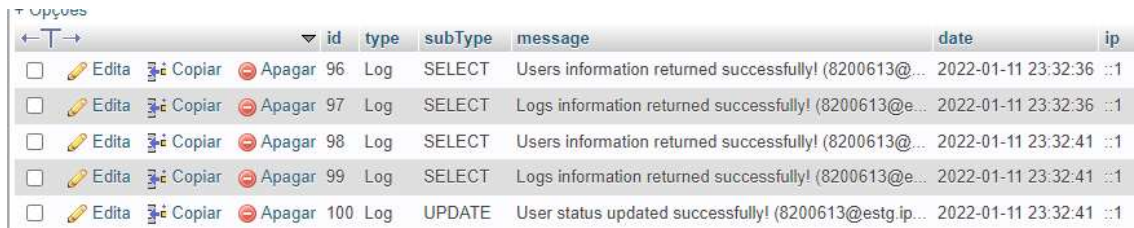
```
includes > error.php
1 <?php
2 if (isset($_GET['error'])) {
3     switch ($_GET['error']) {
4         case 'inputs':
5             $message = 'Please fill in all the required fields.';
6             break;
7         case 'firstname':
8             $message = 'Please enter a valid first name.';
9             break;
10        case 'lastname':
11            $message = 'Please enter a valid last name.';
12            break;
13        case 'telephone':
14            $message = 'Please enter a valid telephone.';
15            break;
16        default:
17            $message = 'Something went wrong!';
18            break;
19    }
20 }
21
22 <div class="alert alert-danger alert-dismissible fade show text-center" role="alert">
23     <i class="bi bi-exclamation-triangle-fill"></i>
24     <strong>Error!</strong> <?php echo ($message); ?>
25     <button type="button" class="btn-close shadow-none" data-bs-dismiss="alert" aria-label="Close"></button>
26 </div>
27 <?php
28 }
29 >
```

Deste modo conseguimos garantir uma maior segurança para a nossa aplicação pois não serão mostrados erros do PHP direto para os utilizadores, podendo expor informação sensível.



## Logs / Erros







Em todas as aplicações os erros são inevitáveis. No entanto podemos armazená-los para uma possível análise. Não só podem ser detetados bugs de software como também possíveis tentativas de ataques informáticos. Implementamos a funcionalidade de registar logs ou erros. Para tal, criamos uma tabela na base de dados específica para armazená-los sempre que seja necessário.



			id	type	subType	message	date	ip
<input type="checkbox"/>	✎ Edita	📋 Copiar	🗑 Apagar	96	Log	SELECT	Users information returned successfully! (8200613@...	2022-01-11 23:32:36 ::1
<input type="checkbox"/>	✎ Edita	📋 Copiar	🗑 Apagar	97	Log	SELECT	Logs information returned successfully! (8200613@e...	2022-01-11 23:32:36 ::1
<input type="checkbox"/>	✎ Edita	📋 Copiar	🗑 Apagar	98	Log	SELECT	Users information returned successfully! (8200613@...	2022-01-11 23:32:41 ::1
<input type="checkbox"/>	✎ Edita	📋 Copiar	🗑 Apagar	99	Log	SELECT	Logs information returned successfully! (8200613@e...	2022-01-11 23:32:41 ::1
<input type="checkbox"/>	✎ Edita	📋 Copiar	🗑 Apagar	100	Log	UPDATE	User status updated successfully! (8200613@estg.ip...	2022-01-11 23:32:41 ::1

Posteriormente os logs poderão ser utilizados pelos administradores, para monitorizarem o desempenho da nossa aplicação

### Logs

ID	Type	Sub Type	Message	Date	IP	
96	Log	SELECT	Users information returned successfully! (8200613@estg.ipp.pt)	11 Jan 2022	::1	
97	Log	SELECT	Logs information returned successfully! (8200613@estg.ipp.pt)	11 Jan 2022	::1	
98	Log	SELECT	Users information returned successfully! (8200613@estg.ipp.pt)	11 Jan 2022	::1	
99	Log	SELECT	Logs information returned successfully! (8200613@estg.ipp.pt)	11 Jan 2022	::1	
100	Log	UPDATE	User status updated successfully! (8200613@estg.ipp.pt)	11 Jan 2022	::1	
101	Log	SELECT	Users information returned successfully! (8200613@estg.ipp.pt)	11 Jan 2022	::1	

## Bloqueio de Utilizadores

O bloqueio de utilizadores é uma funcionalidade que visa restringir o acesso do utilizador sempre que assim seja necessário. Esta funcionalidade apenas pode ser utilizada por utilizadores administradores através do painel de administração. Assim caso o utilizador esteja a prejudicar o bom funcionamento da aplicação ou mesma pela necessidade de aplicar sanções, este poderá ver a sua conta bloqueada e ser-lhe-á negado o acesso.

SAW

Account






My Products

Sell

Dashboard

Sign Out

### Users



















Image	ID	Name	Telephone	Email	Level	Status	Since	Actions
	1	Sérgio Félix	916275619	8200615@estg.ipp.pt	Admin	Allowed	21 Dec 2021	<div><div>✕</div><div>🗑</div></div>
	2	André Pinto	918133838	8200613@estg.ipp.pt	Admin	Allowed	21 Dec 2021	<div><div>✕</div><div>🗑</div></div>
	5	Rui Silva	919191919	ruisilva@gmail.com	User	Allowed	10 Jan 2022	<div><div>✕</div><div>🗑</div></div>
	6	Manuel Alegre	964596486	manuelalegre@outlook.pt	User	Allowed	10 Jan 2022	<div><div>✕</div><div>🗑</div></div>
	7	Joana Albuquerque	939393939	joanaalbuquerque@gmail.com	User	Allowed	10 Jan 2022	<div><div>✕</div><div>🗑</div></div>



## Upload de Imagens

No que toca a upload de imagens, também implementamos algumas técnicas para evitar o acesso fácil ao conteúdo. Sendo imagens publicas, o upload é realizado para uma pasta publica da nossa servidora, no entanto, esta não é indexada. Ou seja, caso o utilizador aceda diretamente á pasta, não lhe será mostrado todos o conteúdo das imagens. Apenas terá acesso caso saiba o link exato da mesma. Mas não é só, todas as imagens são renomeadas antes de serem guardadas. Assim, geramos um nome aleatório para cada imagem, dificultando a obtenção da mesma.

Como podemos verificar, todas as imagens têm um nome aleatório, o que será muito difícil para uma mal-intencionado ter acesso a todas imagens de uma forma fácil.

Nome
 59245697561c20b0f2b40d1.46712810.jpg
 60297361061c20f3bcdcd80.96708351.jpg
 62417297761c20f60190944.17642952.jpg
 78505632461c20aabc3f797.84077461.jpg
 79636370461c20f3bfc004.09009411.jpg
 102129248961c20db441a047.52231934.jpg
 117607394761c20aabc56fa6.70406089.jpg
 121585311961c20b0f29f059.57664631.jpg
 124785509861c20ae475f682.04940147.jpg
 134239085961c20db4412585.78014182.jpg
 141990013861c20aabc4bea5.10736946.jpg
 159711579561c20f601a5ae8.62157380.jpg
 161594856461c20db4424408.39443926.jpg
 163284207661c20f3bcd5d96.68166925.jpg
 165838050161c20ae4768ad2.19448274.jpg
 168875523661c20f601aaa00.07213128.jpg
 184590936961c20b0f2a7e33.36418526.jpg
 201701900561c20ae477ba39.16347448.jpg

## Conclusão

Neste trabalho, foi desenvolvido uma aplicação web de venda de produtos usados. Nela, para além das funcionalidades básicas, implementamos várias soluções para aumentar a segurança da nossa aplicação de modo a efetivar o que foi aprendido nas aulas lecionadas.

Graças á nossa dedicação, foi possível cumprir todos os objetivos idealizados para a realização deste projeto e melhorar os nossos conhecimentos sobre a área de segurança de aplicações.

Conseguimos organizar todos o código para proporcionar uma maior escalabilidade e organização do mesmo. Implementamos um sistema para navegação por rotas de modo a evitar repetição de código; sanitização para evitar ataques indesejado; validação para restringir os dados que podemos aceitar; encriptação para evitar o acesso a dados não autorizados entre muitas mais funcionalidades.

Não só crescemos a nível de conhecimento, como também aumentou a nossa sensibilização para esta área. Muitas das vezes o objetivo é ter uma aplicação funcional e não damos o devido valor á segurança. No entanto, não nos interessa ter uma aplicação funcional, caso esta seja facilmente invadida e se torne inútil, ou por outro lado, o vazamento de informação. Ambos podem causar grandes prejuízo às empresas, por isso segurança é um dos pontos mais importantes durante o desenvolvimento de aplicações. É importante para nós como também para os utilizadores.

Achamos que foi um projeto bem conseguido, que do qual estamos orgulhosos de todos o processo de aprendizagem e o resultado final.

## Bibliografia

### Apresentações da Disciplina SAW

- Aulas Teóricas
- Apresentações

### Documentação PHP

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

### Pesquisa e Aprendizagem

- <https://pt.stackoverflow.com/>
- <https://www.youtube.com/>
- <https://www.udemy.com/pt/>
- <https://github.com/>