SAW – Trabalho Prático

segurança das Aplicações web

André Pinto (8200613) / SERGIO Félix (8200615)

2022

Índice

[Introdução 2](#_Toc92212434)

[Requisitos 2](#_Toc92212435)

[Arquitetura do projeto 3](#_Toc92212436)

# 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.

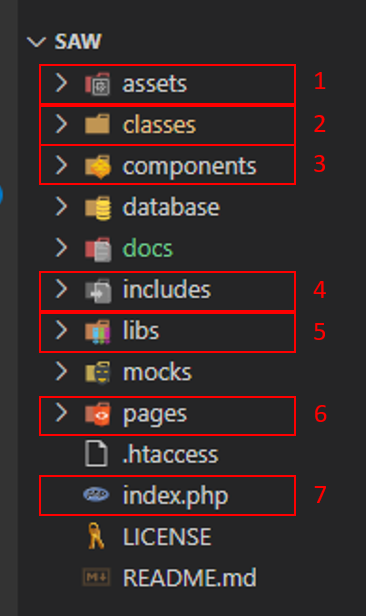
# Arquitetura do Projeto

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**

Uma imagem com texto

Descrição gerada automaticamenteO 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)

**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.

Uma imagem com texto

Descrição gerada automaticamenteUma imagem com texto

Descrição gerada automaticamenteConsoante o URL, a aplicação carrega páginas diferentes, permitindo a navegação sem ter de repetir o Navbar e o Footer.

# Funcionalidades

## Upload de Imagens

# Implementação de Segurança

A implementação de sistemas de segurança foi o principal objetivo deste trabalho. Para tal, implementamos vário conceito aprendido durante as aulas.

## 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.

Uma imagem com texto

Descrição gerada automaticamenteEste procedimento é incluído em todos os inputs do utilizador.

## 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.

Uma imagem com texto

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

Uma imagem com texto

Descrição gerada automaticamente

## Atempt Login

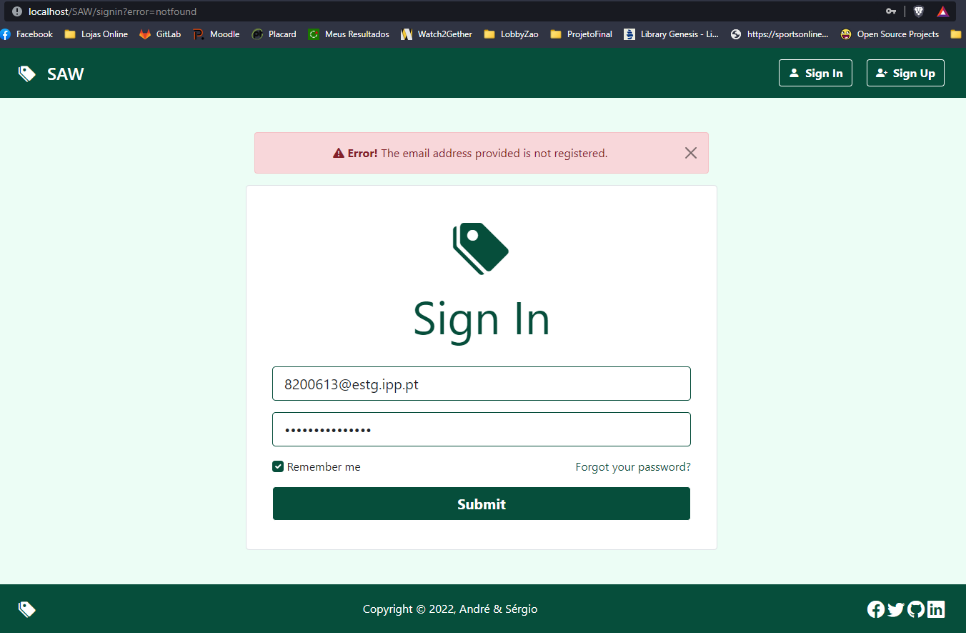
## Recover Password

## Error Handling

Uma imagem com texto

Descrição gerada automaticamenteUma imagem com texto

Descrição gerada automaticamente 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.

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

## Armazenamento