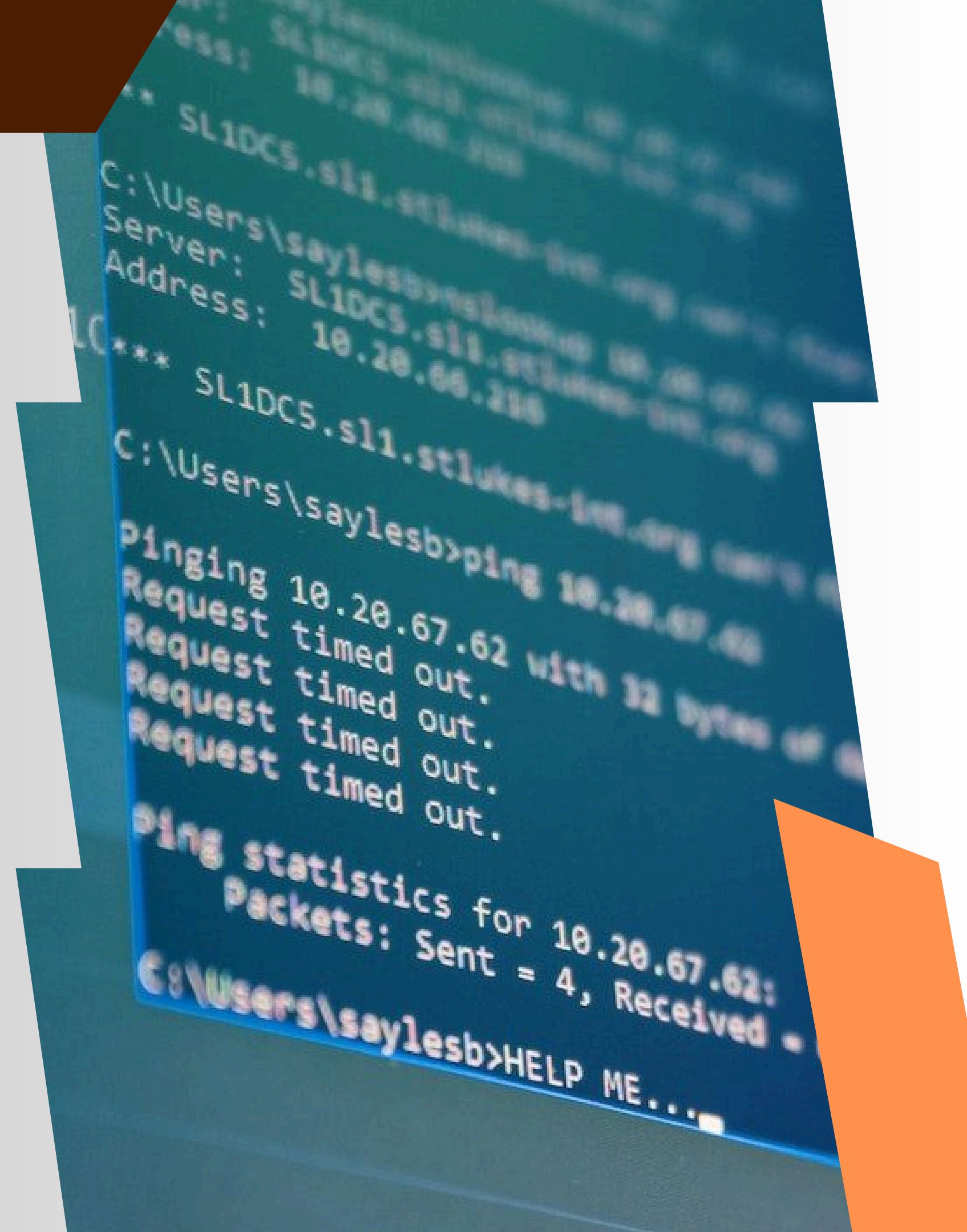


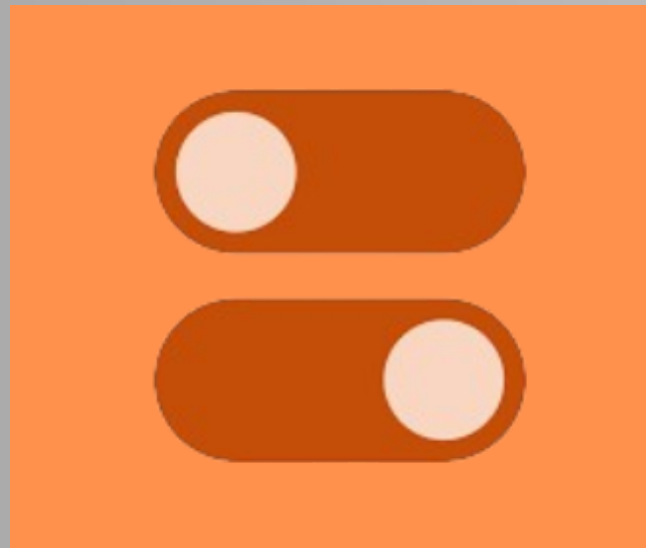
DELEÇÃO EM CASCATA NO MYSQL

Entendendo a deleção automatica de dados
associados



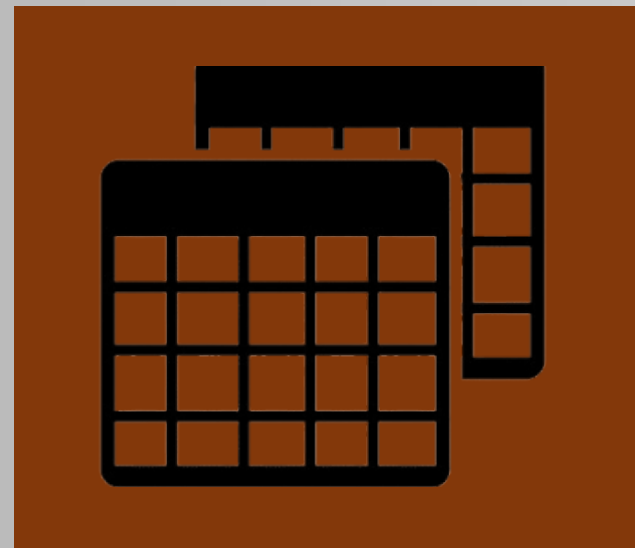
DELEÇÃO EM CASCATA NO MYSQL

Entendendo a deleção automatica de dados associados



Contexto da Deleção em Cascata

No sistema, existem tabelas relacionadas, como usuarios, codigos_verificacao e codigos_redefinicao, onde a deleção de um usuário deve automaticamente excluir os dados associados.



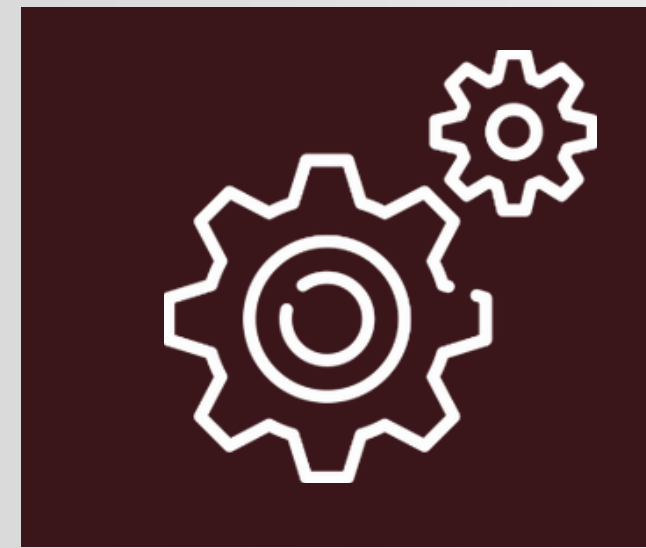
Relacionamento entre Tabelas

As tabelas codigos_verificacao e codigos_redefinicao possuem relacionamentos com a tabela usuarios.



Processo de Exclusão

Ao excluir um usuário, as informações relacionadas nessas tabelas são removidas automaticamente.



Configuração da Deleção em Cascata

A deleção em cascata é configurada no banco de dados para que, quando um registro em usuarios for excluído, os registros associados nas tabelas relacionadas também sejam removidos.



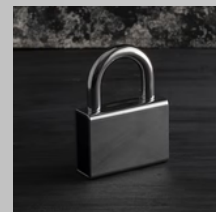
Prevenção de Registros órfãos

Esse processo impede a existência de registros órfãos no banco de dados.

HASH DE SENHA



HASH DE SENHA



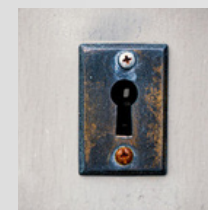
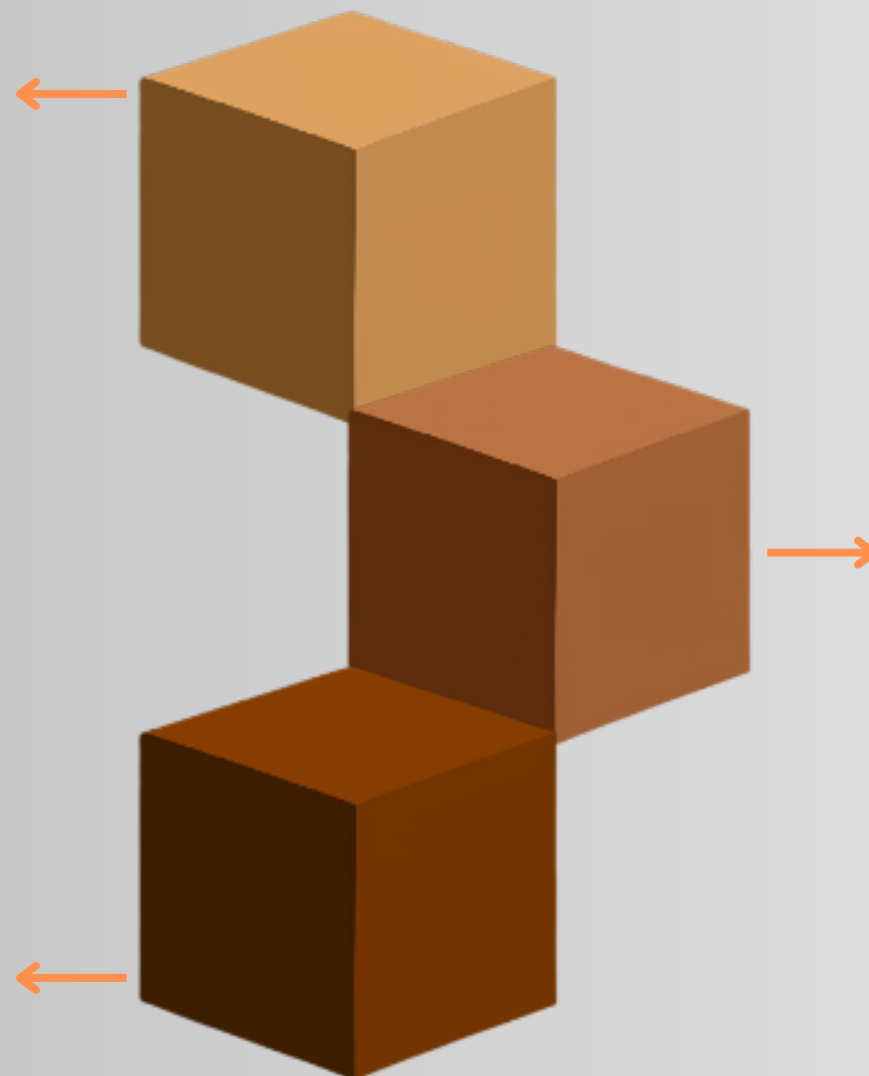
Armazenamento seguro de senhas

As senhas dos usuários são armazenadas de maneira segura utilizando um algoritmo de hash, impedindo que senhas em texto simples sejam guardadas no banco de dados.



Armazenamento do hash

O hash gerado é armazenado na tabela 'usuarios' do banco de dados, ao invés de guardar a senha em texto claro.



Convertendo para HASH

Quando um novo usuário se registra, a senha fornecida é convertida em um hash usando a função `password_hash()`.

VANTANGENS DO HASH DE SENHA



Segurança

Armazenar senhas como hash protege as informações sensíveis contra roubo, pois o hash é uma representação irreversível da senha.

Reforço com Algoritmo Seguro

O sistema usa o algoritmo PASSWORD_BCRYPT, que é resistente a ataques de força bruta e é recomendado por ser seguro e eficiente.

SESSIONS

Contexto das Sessões

O sistema utiliza sessões para gerenciar a autenticação do usuário e manter o estado entre as páginas durante a navegação.

Função `session_start()`

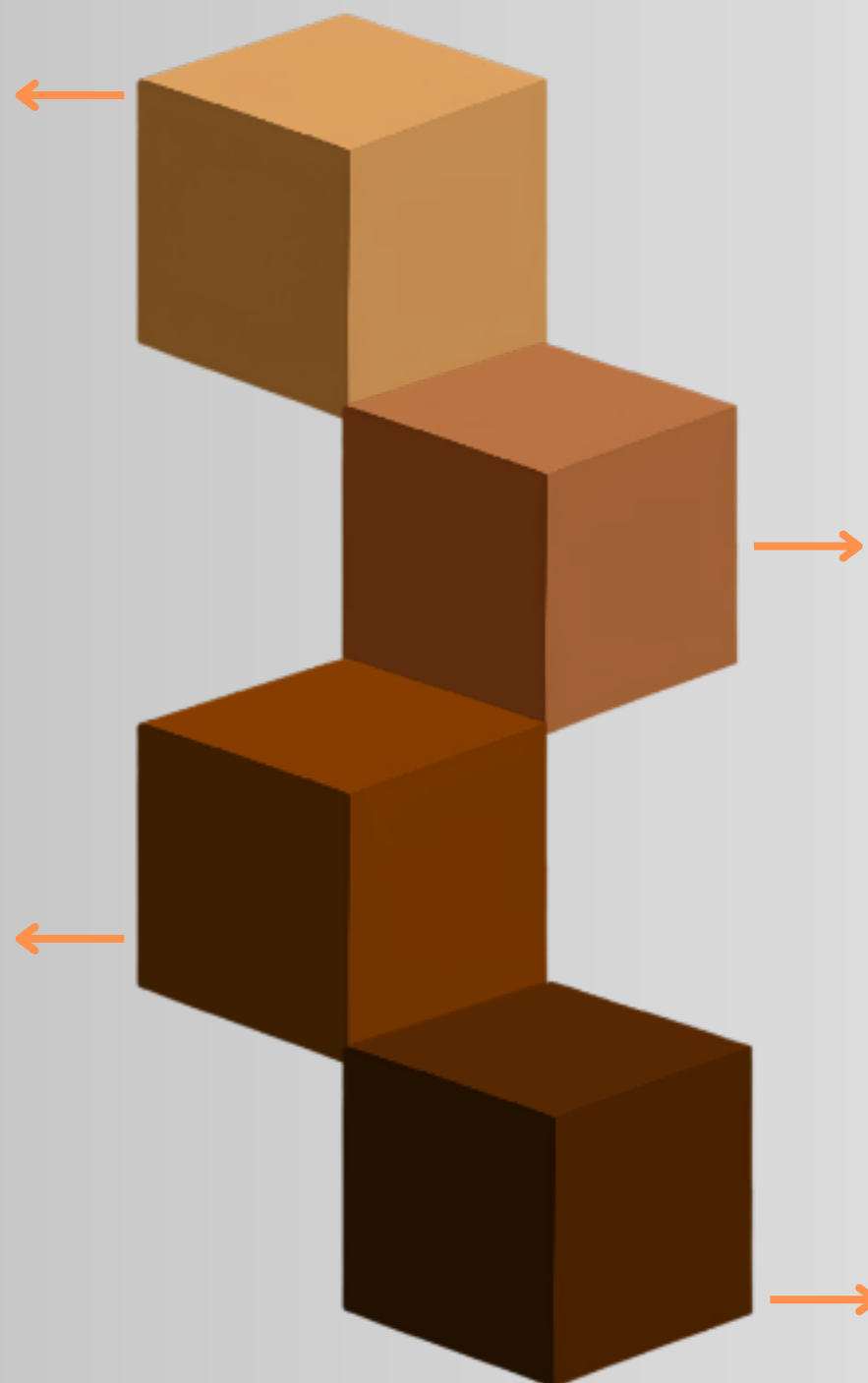
Essa função inicia uma nova sessão ou retoma uma sessão existente, permitindo que os dados da sessão sejam acessados.

Início da Sessão

No início de cada página que precisa de informações de sessão, como `login.php` e `aplicacao.php`, a função `session_start()` é chamada.

Finalização da Sessão

No arquivo `logout.php`, a função `session_destroy()` é chamada para finalizar a sessão.



COOKIES

Armazenamento de informações

Os cookies armazenam informações no navegador, permitindo que o sistema lembre o usuário.



Personalização da experiência

Os cookies ajudam na personalização da experiência do usuário, guardando preferências e identificadores de sessão.

Manutenção do estado de autenticação

Facilitam a manutenção do estado de autenticação entre sessões, como manter o usuário logado.

FUNÇÕES PHP

Funções de Processamento de Dados

PHP é utilizado para processar e validar os dados enviados pelo formulário, como no arquivo registro.php, onde os dados do usuário são validados e armazenados no banco de dados.



Funções de Autenticação e Sessão

PHP gerencia a autenticação do usuário, utilizando `$_SESSION` para armazenar dados do login e realizar verificações de autenticação, como nos arquivos login.php e logout.php.

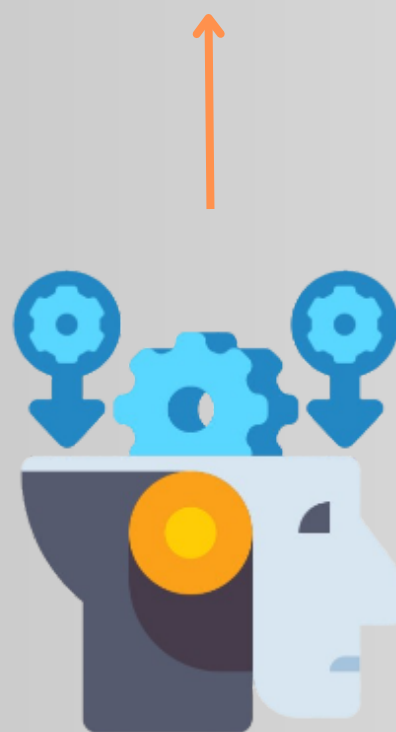
Hash de Senha

A função `password_hash()` é usada para gerar um hash seguro da senha do usuário, garantindo que ela seja armazenada de forma segura no banco de dados.

FUNÇÕES JAVASCRIPT

Validação de Formulários

Funções JavaScript são usadas para validar os dados antes de serem enviados ao servidor, garantindo que os campos obrigatórios estejam preenchidos e que os formatos, como e-mail e senha, sejam corretos.



Verificação de Força de Senha

Usamos uma função em JavaScript para verificar a força da senha enquanto o usuário a digita, analisando critérios como comprimento da senha, presença de números, caracteres especiais e letras maiúsculas/minúsculas, fornecendo feedback imediato sobre a segurança da senha.

Exibição Dinâmica

JavaScript pode ser usado para mostrar ou ocultar elementos da página sem recarregar, como exibir mensagens de erro ou sucesso ao registrar/login, ou fornecer feedback sobre os dados do formulário.

INCLUDE E REQUIRE NA APLICAÇÃO

Include

Inclui o conteúdo de outro arquivo PHP no script atual.

Se o arquivo não for encontrado, o PHP gera um aviso.

O script continuará executando após o aviso como usamos nos arquivos :
index.php, login.php
ou perfil.php

Require

Funciona de forma semelhante ao include.

Se o arquivo não for encontrado, o PHP gera um erro fatal. Interrompe a execução do script ao encontrar um erro como usamos nos arquivos : login.php, registro.php, verificar.php

CONEXÃO MYSQL

Função da Conexão MySQL

Estabelece a comunicação entre o sistema PHP e o banco de dados MySQL, permitindo a execução de consultas e manipulação de dados.



Arquivo de Gerenciamento

O arquivo `conexao.php` é responsável por gerenciar a conexão com o banco de dados.

Uso de prepared statements

A conexão é feita de forma segura utilizando prepared statements para evitar injeções SQL.

CONEXÃO MYSQL

Uso do mysqli

O mysqli é utilizado para conectar ao banco de dados e realizar operações.

Tratamento de Erros

Se a conexão falhar, o sistema retorna um erro informando o problema de conexão.



Código de Conexão

O código de conexão utiliza localhost, usuário, senha e nome do banco para estabelecer a conexão.

Exemplo de Uso

Arquivos PHP que interagem com o banco de dados, como registro.php, login.php ou verificar.php, o arquivo conexao.php deve ser incluído.

TRY E CATCH

Execução do código

O bloco 'try' executa o código que pode gerar erro.

Controle de falhas

Controla falhas sem interromper o sistema.

Uso na aplicação

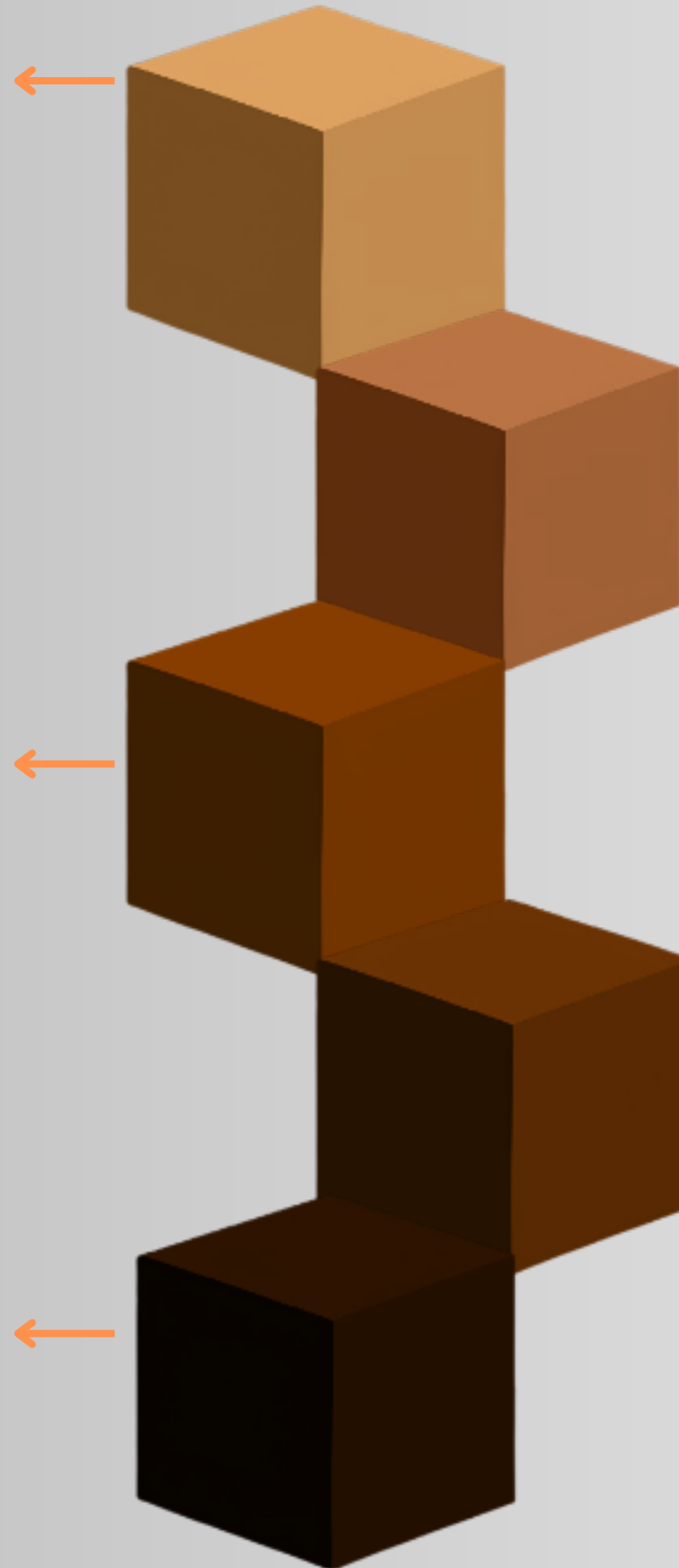
no arquivo mail.php usamos o try e catch para tentar pegar o email de onde vai ser enviado o código de verificação e se caso não conseguimos mostra uma mensagem de erro

Captura de erros

O bloco 'catch' captura e trata o erro, exibindo mensagens amigáveis e evitando a interrupção do sistema.

Experiência do usuário

Melhora a experiência do usuário com mensagens de erro claras.



ESTRUTURAS IF , ELSEIF E ELSE

1 Controle de fluxo do programa

Controlam o fluxo do programa, executando blocos de código baseados em condições específicas.

4 Uso nos Arquivos: login.php

No arquivo login.php, a estrutura if verifica se o email e a senha estão corretos, autenticando o usuário.

2 Uso nos Arquivos: verificar.php

O arquivo verificar.php utiliza a estrutura if para verificar se o código de verificação e o email fornecido existem no banco.

5 Uso do elseif em login.php

O elseif verifica se o campo de email ou senha está vazio, exibindo uma mensagem de erro.

3 Uso do else em verificar.php

Em arquivos PHP que interagem com o banco de dados, como registro.php, login.php ou verificar.php, o arquivo conexao.php deve ser incluído.

6 Uso do else em login.php

Se as credenciais estiverem incorretas, o else exibe uma mensagem informando que o login falhou.

REFERÊNCIAS COM -> EM PHP

Uso do operador -> na programação orientada a objetos

1 Função do operador ->

O operador -> é utilizado em PHP para acessar propriedades ou métodos de objetos, sendo essencial na programação orientada a objetos (POO).

4 Método execute()

`$conexao->execute()` executa a consulta SQL previamente preparada.

2 Uso em arquivos PHP

No arquivo `conexao.php`, o operador -> é utilizado para acessar métodos de conexão do MySQL.

5 Objeto de conexão

`$conexao` representa um objeto de uma classe que gerencia a conexão com o banco de dados.

3 Método prepare()

`$conexao->prepare()` prepara a consulta SQL para execução.

ESTRUTURAS WHILE EM PHP

Continuação da execução

A execução continua até que a condição se torne falsa.

Execução do bloco de código

Se a condição for verdadeira, o código dentro do bloco `{ }` é executado.

Uso na aplicação

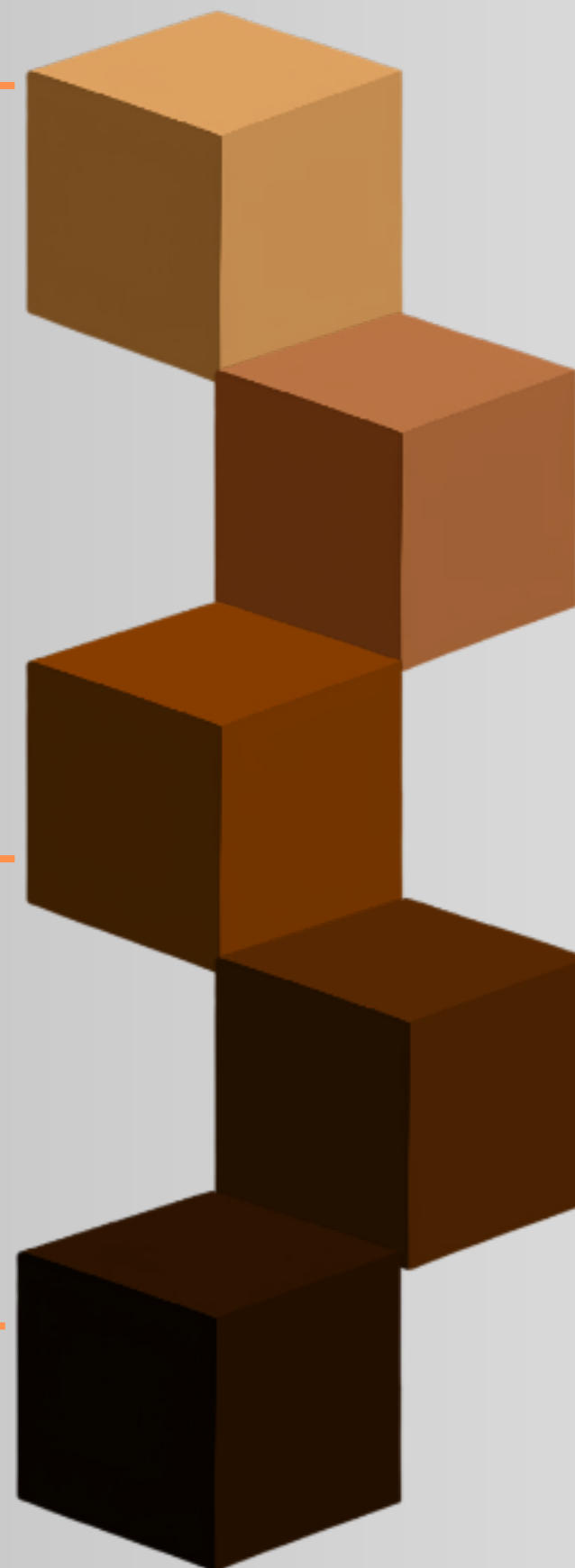
No arquivo `aplicação.php` usamos `while` para filtrar os assuntos das perguntas do quiz

Função da estrutura while

A estrutura de repetição `while` executa um bloco de código enquanto uma condição for verdadeira.

Avaliação da condição

A expressão dentro do `while` é avaliada antes de cada iteração.



MÉTODOS POST EM PHP

1

Método POST

O método POST é usado para enviar dados de um formulário HTML para o servidor.

2

Segurança dos Dados

Ele envia os dados de forma segura, ocultando-os na URL.

3

Uso nos Arquivos

O método POST é amplamente utilizado em diferentes arquivos para processar dados de formulários.

4

login.php e esqueceu_senha.php

Nesses arquivos, o método POST é utilizado para receber dados como o email e a senha do usuário, a fim de processar e validar a autenticação.

5

registro.html

Neste arquivo, o método POST é usado para enviar dados ao servidor, como o email e a senha, para o processamento de registro de usuário.

CRUD COMPLETO

Create

Em registro.php, a função de cadastro de novos usuários cria um novo registro no banco de dados usando dados enviados através do formulário. Isso é feito com uma consulta SQL INSERT.

Read

Em login.php, o sistema verifica se o usuário existe ao consultar o banco de dados, utilizando uma consulta SQL SELECT para verificar se o email e a senha coincidem.

Update

Em verificar.php, a função de verificação de código de email pode atualizar o status de verificação do usuário no banco de dados com a consulta SQL UPDATE, alterando o campo verificado.

Delete

Em exclusao_usuario.php, o sistema permite que um usuário seja deletado do banco de dados com uma consulta SQL DELETE.

FINALIDADE DA APLICAÇÃO

Autenticação de usuários

A aplicação permite que os usuários realizem o cadastro e façam login com verificação em duas etapas.

Gerenciamento de usuários

Facilita a administração dos usuários dentro da aplicação.



Interatividade

Os usuários podem participar ativamente do quiz, promovendo engajamento.

Funcionalidade de Quiz

Oferece um quiz interativo com perguntas sobre diversos temas.

VERIFICAÇÃO EM DUAS ETAPAS (2FA)

Uma camada extra de segurança

Objetivo da 2FA

A verificação em duas etapas (2FA) tem como objetivo adicionar uma camada extra de segurança à aplicação.



Proteção contra acesso indevido

Impede o acesso não autorizado, mesmo que alguém tenha obtido a senha do usuário.



Acesso autorizado

Garante que apenas o usuário autorizado consiga acessar sua conta.

COMO FUNCIONA A VERIFICAÇÃO EM DUAS ETAPAS

Passo 1

Cadastro e Login

O usuário se cadastra normalmente com um email e senha.

Envio de Código de Verificação

Após o login, um código de verificação é enviado para o email do usuário.

Passo 2

Passo 3

Validação do Código

O código enviado é armazenado temporariamente no banco de dados.

Autenticação Completa

Só após a validação do código, o usuário tem acesso completo à sua conta.

Passo 4

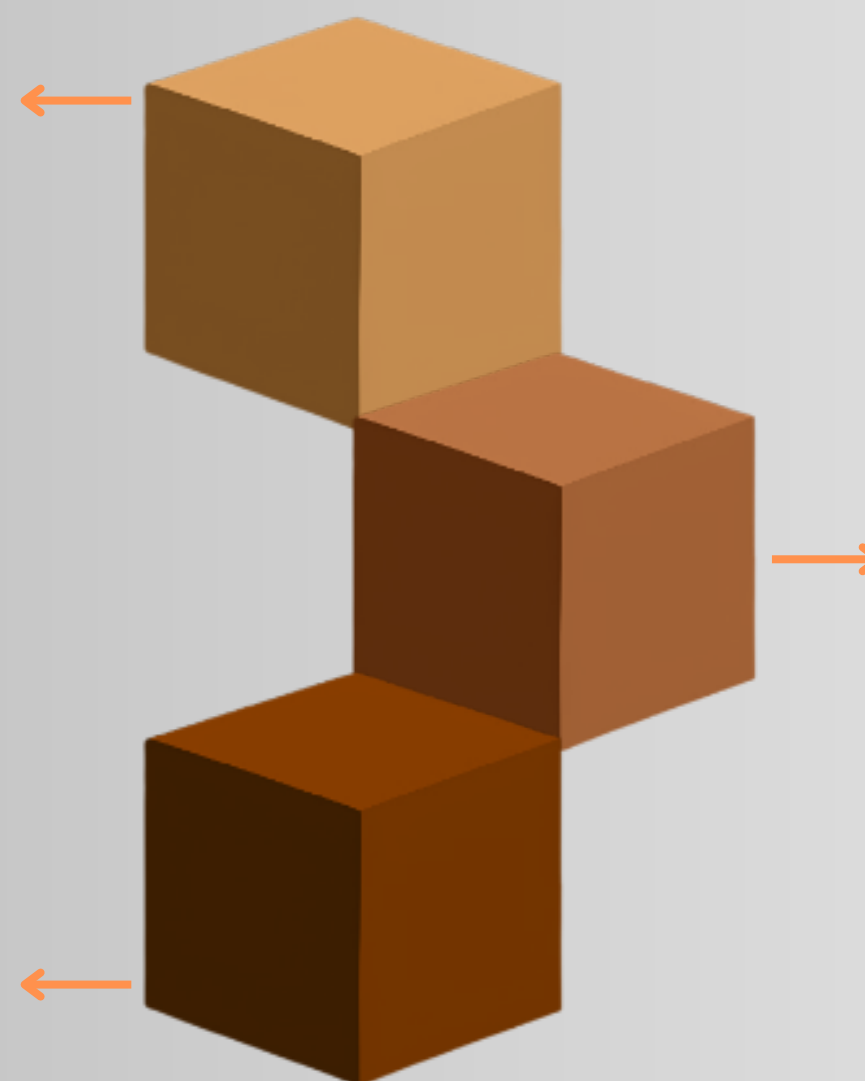
PROCESSO DE VERIFICAÇÃO EM DUAS ETAPAS NOS ARQUIVOS

Arquivo `codigos_verificacao.php`

Gerencia o envio de código para o email do usuário e o registro desse código no banco de dados.

Arquivo `aplicacao.php`

Após o login, o código de verificação é verificado aqui, garantindo que o acesso seja autorizado.



Arquivo `login.php`

Responsável por capturar as credenciais do usuário e iniciar o processo de verificação de duas etapas.

ROTAS

Tela inicial da aplicação (index.html)

Bem-vindo ao Sistema de Autenticação

Escolha uma opção abaixo para acessar sua conta ou criar uma nova.

Registrar

Login

ROTAS

Registro (Registro.html)

Registro de Usuário

E-mail:

Senha:

Confirmar Senha:

Registrar

[Home](#)

ROTAS

Login (Login.html)

Login

E-mail:

Senha:

Entrar

[Esqueceu sua senha?](#)

[Home](#)

ROTAS

esqueceu a senha (esqueceu_senha.html)

Esqueceu sua senha?

Digite seu e-mail para receber um código de redefinição de senha.

E-mail:

Enviar código

[Home](#)

ROTAS

Arquivo index.html

A rota inicial da aplicação. A partir daqui, o usuário é redirecionado para login ou registro

Arquivo login.html e login.php

Responsáveis pela rota de login. O formulário de login na página login.html envia as informações para login.php, que valida as credenciais e redireciona o usuário

Arquivo registro.html e registro.php

Rota de registro de novos usuários. O formulário coleta dados do usuário, e registro.php trata o envio e armazena as informações no banco de dados

Arquivo esqueceu_senha.html e esqueceu_senha.php

Gerenciam a rota de recuperação de senha. O usuário informa o email para receber um link de redefinição de senha.

Arquivo aplicação.php

Controla a principal lógica de interação após o login bem-sucedido como verificação de dois fatores (2FA)

Arquivo pagina_principal.php

A rota principal para a navegação após o login, onde o usuário pode acessar diversas funcionalidades da aplicação

Arquivo logout.php

Rota de logout que encerra a sessão do usuário e redireciona para a página de login



Obrigado por assistir professor

desculpa ter ficado muito grande

Feito por :Gustavo Leandro



ps : me da nota boa

