|  |
| --- |
| DIGITAL  close-up van handen met een schaar en stiften, aan het werkSCRAMBLEPAD |

|  |  |  |
| --- | --- | --- |
| Geïntegreerde proef 2022-2023 |  |  |
| André da Cruz Ribeiro T.I Don Bosco Hoboken |  |  |

Inhoudsopgave

[Inleiding 3](#_Toc126576732)

[Software 4](#_Toc126576733)

[PHP CODE 4](#_Toc126576734)

[DATABASE 4](#_Toc126576735)

[JAVASCRIPT 4](#_Toc126576736)

[HTML 4](#_Toc126576737)

[CSS 4](#_Toc126576738)

[ARDUINO 4](#_Toc126576739)

[Hardware 5](#_Toc126576740)

[WT32-ETH01 5](#_Toc126576741)

[SLOTEN 5](#_Toc126576742)

[BESLUIT 5](#_Toc126576743)

[BIBLIOGRAFIE 5](#_Toc126576744)

[LOGBOEK 5](#_Toc126576745)

# voorwoord

Ik presenteer mijn Geïntegreerde Proef over digitale scramblepads als veilig alternatief voor traditionele sloten. In mijn GIP bespreek ik de ontwikkeling van digitale scramblepads.

Ik heb dit project gekozen vanwege de voordelen van digitale scramblepads ten opzichte van traditionele sloten, zoals verbeterde veiligheid, gebruiksgemak en flexibiliteit.

Graag wil ik ook de heer Van Reck en de heer Thijs, die mij gedurende dit project hebben begeleid en waardevolle feedback hebben gegeven.

Ook wil ik graag mijn dank uitspreken aan al mijn klasgenoten die hebben bijgedragen aan de totstandkoming van mijn GIP.

# Inleiding

Het doel van mijn GIP is om een veiliger keypad te ontwerpen voor mensen die hun deuren willen beveiligen met een code. Mijn keypad verschilt van de normale keypads doordat de cijfers elke keer van plaats veranderen nadat er een cijfer wordt ingedrukt. Zo kan niemand de code afkijken of raden op basis van de volgorde van de toetsaanslagen

Ik heb de volgende onderdelen gebruiken:

* 2 ESP-32 WROM-DEV
* 2 elektronische sloten
* 2 RGB LEDS

# Software

## Index

#### Index

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta http-equiv="X-UA-Compatible" content="IE=edge" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<title>Home Page</title>

<style>

/\* Set body margin and padding to 0 to remove default spacing \*/

body {

margin: 0;

padding: 0;

overflow: hidden;

}

/\* Style the buttons \*/

button {

height: 100%;

border: none;

outline: none;

cursor: pointer;

font-size: 2em;

font-weight: bold;

color: white;

background-color: transparent;

transition: all 0.3s cubic-bezier(0.25, 0.8, 0.25, 1);

}

/\* Style the links inside the buttons \*/

a {

display: block;

text-align: center;

text-decoration: none;

color: inherit;

}

/\* Center the text inside the button and allow clicking anywhere \*/

button a {

display: flex;

justify-content: center;

align-items: center;

height: 100%;

width: 100%;

}

/\* Center the link text inside each button \*/

.left a,

.right a {

display: flex;

justify-content: center;

}

/\* Style the left button \*/

.left {

width: 50%;

float: left;

background-color: green;

}

/\* Style the right button \*/

.right {

width: 50%;

float: right;

background-color: blue;

}

/\* Add a hover effect to the buttons \*/

button:hover {

transform: scale(1.1);

transition: all 0.3s ease-in-out;

}

/\* Adjust hover effect on the left button \*/

.left:hover {

transform: scale(1.1);

transition: all 0.3s ease-in-out;

}

/\* Position the button container \*/

.container {

position: absolute;

top: 0;

bottom: 0;

width: 100%;

}

</style>

</head>

<body>

<!-- Create a container for the buttons -->

<div class="container">

<!-- Create two buttons that split the screen horizontally -->

<button class="left"><a href="admin/admin.html">admin page</a></button>

<button class="right"><a href="setup.html">setup page</a></button>

</div>

</body>

</html>

Dit HTML-bestand (index.html) is bedoeld als de startpagina voor een toepassing die een digitale scramblepad gebruikt om deuren te beveiligen. De pagina bevat twee knoppen, een groene en een blauwe, die elk de gebruiker naar een andere pagina leiden.

De groene knop geeft toegang tot de beheerderspagina waar de beheerder nieuwe gebruikers en deuren kan toevoegen of verwijderen en ook de database kan aanmaken.

De blauwe knop leidt naar de instellingspagina waar de beheerder de naam van de deur kan kiezen waarvoor de scramblepad wordt gebruikt.

De code bevat ook CSS-styling om de lay-out van de pagina te bepalen en de knoppen te stijlen. Er wordt gebruik gemaakt van flexbox om de tekst in de knoppen te centreren en overgangseffecten toe te voegen wanneer de gebruiker de muis over de knoppen beweegt. Het doel is om de pagina gemakkelijk te navigeren en de gebruiker de nodige opties te bieden om de digitale scramblepad te configureren en te gebruiken.

## Login

#### Setup

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta http-equiv="X-UA-Compatible" content="IE=edge" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<title>Setup Page</title>

</head>

<body>

<form action="login/scramblepad.html" onsubmit="setCookies()">

<input

type="text"

name="deurnaam"

id="deurnaam"

placeholder="Deur Naam"

required

/>

<select name="deurrank" id="deurrank">

<option selected>Kies rank</option>

<option value="1">1</option>

<option value="2">2</option>

<option value="3">3</option>

<option value="4">4</option>

</select>

<button type="submit">Finish Setup</button>

</form>

<script>

function setCookies() {

// Get the values of the input and select fields

let deurnaamValue = document.getElementById("deurnaam").value.toLowerCase();

let deurrankValue = document.getElementById("deurrank").value.toLowerCase();

// Set the cookies with the values

document.cookie = "deurnaam=" + encodeURIComponent(deurnaamValue);

document.cookie = "deurrank=" + encodeURIComponent(deurrankValue);

}

</script>

</body>

</html>

Deze code is de HTML en JavaScript code voor de "Setup Page". Het bevat een formulier met drie invoervelden: een tekstveld voor de naam van de deur, een dropdown-menu voor de rank van de deur en een knop om de setup te voltooien. Wanneer de gebruiker op de knop klikt, wordt de JavaScript-functie "setCookies()" opgeroepen, die de waarden van de invoervelden ophaalt en deze waarden gebruikt om cookies te maken voor de naam en de rang van de deur. Deze cookies zullen later gebruikt worden door de "verify.php" pagina, waar de rang van de gebruiker wordt vergeleken met de rang van de deur. Op basis hiervan kan worden bepaald of de gebruiker toestemming heeft om de deur te openen.

#### Scramblepad

###### HTML en JS

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta http-equiv="X-UA-Compatible" content="IE=edge" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<link rel="stylesheet" href="scramblepad-style.css" />

<link

href="https://fonts.googleapis.com/css2?family=Bebas+Neue&display=swap"

rel="stylesheet"

/>

<title>test page for scramblepad</title>

</head>

<body>

<div class="parent">

<form action="verify.php" method="post">

<input type="hidden" name="password" id="password" value="" required />

<div class="username-text-field">

<input

class="username-input border"

name="username"

id="username"

placeholder="Username"

aria-label="Username"

aria-describedby="Username"

type="text"

required

/>

</div>

<div class="button-row">

<button class="button border" type="button"></button>

<button class="button border" type="button"></button>

<button class="button border" type="button"></button>

</div>

<div class="button-row">

<button class="button border" type="button"></button>

<button class="button border" type="button"></button>

<button class="button border" type="button"></button>

</div>

<div class="button-row">

<button class="button border" type="button"></button>

<button class="button border" type="button"></button>

<button class="button border" type="button"></button>

</div>

<div class="button-row">

<button class="button border no-invert ❌" type="button">❌</button>

<button class="button border" type="button"></button>

<button class="button border no-invert ✔" type="submit">✔</button>

</div>

</form>

</div>

</body>

</html>

<script>

function shuffleArray(arr) {

arr.sort(() => Math.random() - 0.5);

}

let arr = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9];

shuffleArray(arr);

getKeys();

function getKeys() {

let key = 0;

document.querySelectorAll('button[type="button"]').forEach((element) => {

if (element.textContent == "❌") return;

element.textContent = arr[key];

key++;

});

}

document.querySelector("form").addEventListener("click", (event) => {

if (event.target.textContent == "✔") return;

if (event.target.textContent == "❌") {

document.getElementById("password").value = "";

return;

}

document.getElementById("password").value += event.target.textContent;

shuffleArray(arr);

getKeys();

});

</script>

###### CSS

\* {

box-sizing: border-box;

}

body {

overflow: hidden;

font-family: "Bebas Neue", cursive;

font-size: large;

background-color: rgb(37, 39, 42);

}

.parent {

display: grid;

place-content: center;

height: 100vh;

}

.border {

border-radius: 20px;

border-width: 2px;

border-style: solid;

border-color: black;

font-size: 1.5em;

}

.username-text-field {

padding: 10px;

}

.username-input {

width: 60.5vw;

height: 6vh;

}

.button-row {

padding: 10px;

}

.button {

position: relative;

overflow: hidden;

width: 20vw;

height: 10vh;

text-align: center;

line-height: 50px;

}

.button::before {

content: "";

position: absolute;

top: 0;

left: 0;

width: 100%;

height: 100%;

background-color: transparent;

filter: invert(1);

}

button:not(.no-invert):hover::before {

background-color: rgb(37, 39, 42);

}

.❌ {

background-color: tomato;

}

.❌:hover {

background-color: red;

}

.✔ {

background-color: greenyellow;

}

.✔:hover {

background-color: green;

}

###### Uitleg

Dit is de webpagina waarop de gebruiker zijn code kan invoeren. De knoppen bevatten cijfers van 0 tot 9 die bij elke invoer in een willekeurige volgorde worden geschud.

De code van deze webpagina bestaat uit HTML, CSS en JavaScript. De HTML-code definieert de structuur van de webpagina en de CSS-code geeft de webpagina stijl en opmaak. De JavaScript-code zorgt ervoor dat de knoppen worden geschud.

De JavaScript-code werkt als volgt: wanneer de pagina wordt geladen, wordt er een array gemaakt met de cijfers van 0 tot 9. Vervolgens worden deze cijfers in een willekeurige volgorde geschud met behulp van een functie genaamd shuffleArray(). Deze functie gebruikt de sort() methode van de array met een willekeurige volgorde om de cijfers te schudden.

Vervolgens wordt een functie genaamd getKeys() aangeroepen die de geschudde cijfers toewijst aan de knoppen op de webpagina. Deze functie gebruikt de forEach() methode om door alle knoppen te itereren en de cijfers toe te wijzen op basis van de volgorde van de geschudde array.

Wanneer een gebruiker op een knop klikt, wordt er een event listener geactiveerd die controleert welke knop er is ingedrukt en het bijbehorende cijfer toevoegt aan een inputveld dat is verborgen op de webpagina. De shuffleArray() en getKeys() functies worden vervolgens opnieuw aangeroepen om de knoppen opnieuw te schudden en de nieuwe geschudde cijfers toe te wijzen aan de knoppen.

De CSS-code is verantwoordelijk voor de stijl en opmaak van de webpagina. Het definieert de kleuren, de grootte en de vorm van de knoppen en het inputveld. Het zorgt ook voor animaties en visuele feedback wanneer een gebruiker op een knop klikt.

#### Verificatie

<?php

$link = mysqli\_connect("localhost", "root", "", "locks");

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

$username = $\_POST['username'];

$username = strtolower($username);

$password = $\_POST['password'];

$deurnaam\_cookie = $\_COOKIE['deurnaam'];

$deurnaam\_cookie = strtolower($deurnaam\_cookie);

$deurrank\_cookie = $\_COOKIE['deurrank'];

$sanitized\_username = mysqli\_real\_escape\_string($link, $username);

$sanitized\_password = mysqli\_real\_escape\_string($link, $password);

$sql = "SELECT `rank` FROM `users` WHERE `username`='$sanitized\_username' AND `password`='$sanitized\_password'";

$result = mysqli\_query($link, $sql) or die(mysqli\_error($link));

$user\_rank = mysqli\_fetch\_array($result);

if($user\_rank['rank'] >= $deurrank\_cookie) {

$sql = "UPDATE `doors` SET `status`='open' WHERE `deur naam`='$deurnaam\_cookie'";

$result = mysqli\_query($link, $sql) or die(mysqli\_error($link));

sleep(5); // wait for 5 seconds before updating the status to close

$sql = "UPDATE `doors` SET `status`='close' WHERE `deur naam`='$deurnaam\_cookie'";

$result = mysqli\_query($link, $sql) or die(mysqli\_error($link));

sleep(3);

header("Location: scramblepad.html");

exit();

} else {

// Redirect the user to the scramblepad.html page

header("Location: scramblepad.html");

exit();

}

mysqli\_close($link);

}

?>

Dit stukje PHP code maakt verbinding met een MySQL database genaamd "locks" op de localhost. Vervolgens controleert het of er een POST request is verzonden door de client. Als dat zo is, worden de waarden van de velden "username", "password", "deurnaam" en "deurrank" vanuit de POST data opgeslagen in variabelen. De username en deurnaam worden ook omgezet naar kleine letters om eventuele problemen met hoofdletters te voorkomen.

De functie mysqli\_real\_escape\_string wordt gebruikt om de waarden van de variabelen "username" en "password" te ontsmetten voordat ze worden gebruikt in een MySQL query, om SQL injection aanvallen te voorkomen. Vervolgens wordt er een query uitgevoerd om de rank van de gebruiker op te halen uit de "users" tabel van de database.

Als de rank van de gebruiker gelijk of hoger is dan de rank die is opgeslagen in de "deurrank" cookie, wordt de status van de deur in de "doors" tabel van de database gewijzigd naar "open". Er wordt vervolgens 5 seconden gewacht voordat de status van de deur weer wordt gewijzigd naar "close", dan wordt de gebruiker omgeleid naar de "scramblepad.html" pagina.

Als de rank van de gebruiker lager is dan de rank die is opgeslagen in de "deurrank" cookie, wordt de gebruiker omgeleid naar de "scramblepad.html" pagina.

Als laatste wordt de verbinding met de database gesloten en wordt er geen verdere output gegenereerd.

#### Deur status

## Administrator

#### Index

#### Gebruikers toevoegen

#### Gebruikers verwijderen

#### Deuren toevoegen

#### Database aanmaken

## Esp

# Hardware

# BESLUIT

# BIBLIOGRAFIE

# LOGBOEK