

ZESPÓŁ SZKÓŁ
ELEKTRONICZNYCH
I GIMNAZJUM NR 3
W JELENIEJ GÓRZE



SPRAWOZDANIE

APLIKACJA INTERNETOWA
(*TEMAT ZAJĘĆ*)

ALEKSANDER KOWALSKI
(*AUTOR/AUTORZY*)

JELENIA GÓRA 18.03.19(*DATA*)

ALEKSANDER KOWALSKI
(*AUTOR/AUTORZY*)

OCENA (*WYPEŁNIA PROWADZĄCY*)

Spełnienie wymogów redakcyjnych:

.....

Wykonanie, udokumentowanie oraz opis wykonanych zadań:

.....

Zastosowanie prawidłowego warsztatu programistycznego:

.....

Spis treści

I. Wstęp.....	6
II. Opis Całej bazy danych.....	6
4. Opis bazy danych.....	6
5. Opis relacji.....	7
4. Opis poszczególnych tabel.....	8
1) Tabela azyl_posty.....	8
2) Tabela azyl_zapytania.....	8
3) Tabela forum_kateg.....	8
4) Tabela forum_odpowiedzi.....	9
5) Tabela news.....	9
6) Tabela ranga.....	9
7) Tabela user_bans.....	10
8) Tabela user_login.....	10
9) Tabela user_ranga.....	10
10) Tabela user_spend_time.....	10
11) Tabela_time.....	11
1. Opis uprawnień konkretnych użytkowników do bazy.....	11
2. Wydajność bazy danych i jej bezpieczeństwo.....	12
III. Opis serwisu.....	12
1. Podstawowe informacje na temat serwisu.....	12
2. Spis mechanizmów i ich opis.....	12
IV. Zarządzanie użytkownikami na forum.....	21
1. Panel admina.....	21
2. Panel użytkownika.....	21
V. Spis plików i ich właściwości.....	22
1. Co gdzie jest ?.....	22
2. Spis uprawnień potrzebnych od odczytu pliku.....	23
VI. Podsumowanie.....	23

I. WSTĘP

Celem projektu było zdobycie doświadczenia w wykonywaniu takich projektów na dany termin i zobaczyć ile rzeczy się potrafi a ile nie. Także była to nauka poprzez trening jakim jest pisanie kodu i szukanie informacji na dane zagadnienie.

Ważna informacja:

- W celu umożliwienia przetestowania wszystkich funkcji serwisu polecam się zalogować na admina.

Login: admin

Hasło: test

II. OPIS CAŁEJ BAZY DANYCH

4. OPIS BAZY DANYCH

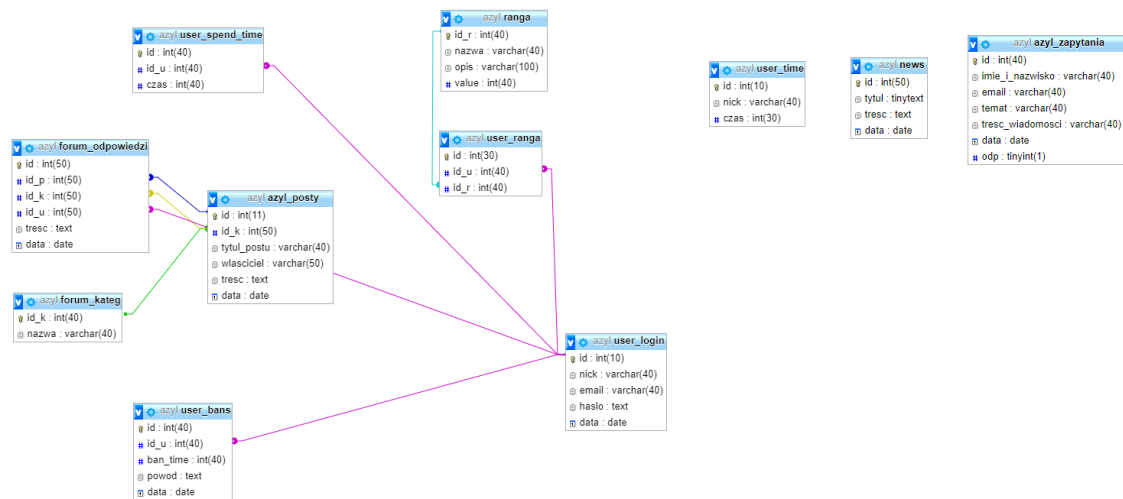
Baza została skonstruowana tak aby nie powtarzały się dane w innych tabelach. Ze względów problemów takie dane tak nick użytkownika musiał się powtórzyć w jednej tabeli ze względu bezpieczeństwa i problemów jakie sprawiało rozbudowane zapytanie do bazy które nie zawsze wykonywało się poprawnie według głównego zamysłu autora.

Tabela	Działanie	Rekordy	Typ	Metoda porównywania napisów	Rozmiar	Nadmiar
azyl_posty	Przeglądaj Struktura Szukaj Wstaw Opróżnij Usuń	5	InnoDB	utf8mb4_general_ci	32 KB	-
azyl_zapytania	Przeglądaj Struktura Szukaj Wstaw Opróżnij Usuń	2	InnoDB	utf8mb4_general_ci	16 KB	-
forum_kateg	Przeglądaj Struktura Szukaj Wstaw Opróżnij Usuń	4	InnoDB	utf8mb4_general_ci	16 KB	-
forum_odpowiedzi	Przeglądaj Struktura Szukaj Wstaw Opróżnij Usuń	3	InnoDB	utf8mb4_general_ci	64 KB	-
news	Przeglądaj Struktura Szukaj Wstaw Opróżnij Usuń	0	InnoDB	utf8mb4_general_ci	16 KB	-
ranga	Przeglądaj Struktura Szukaj Wstaw Opróżnij Usuń	3	InnoDB	utf8mb4_general_ci	16 KB	-
user_bans	Przeglądaj Struktura Szukaj Wstaw Opróżnij Usuń	0	InnoDB	utf8mb4_general_ci	32 KB	-
user_login	Przeglądaj Struktura Szukaj Wstaw Opróżnij Usuń	3	InnoDB	utf8mb4_general_ci	32 KB	-
user_ranga	Przeglądaj Struktura Szukaj Wstaw Opróżnij Usuń	3	InnoDB	utf8mb4_general_ci	64 KB	-
user_spend_time	Przeglądaj Struktura Szukaj Wstaw Opróżnij Usuń	0	InnoDB	utf8mb4_general_ci	32 KB	-
user_time	Przeglądaj Struktura Szukaj Wstaw Opróżnij Usuń	0	InnoDB	utf8mb4_general_ci	16 KB	-
11 tabele	Suma	23	InnoDB	utf8mb4_general_ci	336 KB	0 B

Ilustracja 1: Schemat bazy danych

W każdej tabeli przechowuje konkretne informacje np. azyl_posty przechowuje informacje na temat zapytań na forum.

5. OPIS RELACJI



Ilustracja 2: Schemat relacji

W bazie danych azyl relacje zostały utworzone po to aby uniknąć redundancji (powtarzania się danych) oraz aby móc przeprowadzać analizy na podstawie wielu tabel.

Jak widać na powyższym schemacie trzy tabele nie są używane już tłumacze dlaczego. Ponieważ zabrakło mi czasu aby podłączyć je do funkcji forum które miały działać na starcie. Wolałem się skupić na elementach mechanizmów które miały wady konstrukcyjne lub błędy krytyczne aby je naprawić lub przynajmniej zniwelować do poziomu błędów które są bezpieczne i nie zagrażają wyciekowi danych lub unieruchomienia całego serwisu.

4. OPIS POSZCZEGÓLNYCH TABELEK

1) Tabela azyl_posty

W tej tabeli są przechowywane dane na temat danych postów na forum w konkretnej kategorii (klucz obcy). Są także pola (komórki) które przechowują informacje na temat tytułu danego postu czy kto jest właścicielem danego postu (zapytania)

#	Nazwa	Typ	Metoda porównywania napisów	Atrybuty	Null	Ustawienia domyślne	Komentarze	Dodatkowo
1	id 🗝️	int(11)			Nie	Brak		AUTO_INCREMENT
2	id_k 🗝️	int(50)			Nie	Brak		
3	tytul_postu	varchar(40)	utf8mb4_general_ci		Nie	Brak		
4	wlasciciel	varchar(50)	utf8mb4_general_ci		Nie	Brak		
5	tresc	text	utf8mb4_general_ci		Nie	Brak		
6	data	date			Nie	Brak		

Rysunek 1: Tabela azyl_posty

2) Tabela azyl_zapytania

Ta tabela została utworzona po to aby przechowywać informacje z formularza kontaktowego i móc szybko sprawdzić na jakie zapytania się nie odpowiedziało. Z tej tabeli są później wykorzystywane informacje w innych mechanizmach działania serwisu takich jak: php mailer.

#	Nazwa	Typ	Metoda porównywania napisów	Atrybuty	Null	Ustawienia domyślne	Komentarze	Dodatkowo
1	id 🗝️	int(40)			Nie	Brak		AUTO_INCREMENT
2	imie_i_nazwisko	varchar(40)	utf8mb4_general_ci		Nie	Brak		
3	email	varchar(40)	utf8mb4_general_ci		Nie	Brak		
4	temat	varchar(40)	utf8mb4_general_ci		Nie	Brak		
5	tresc_wiadomosci	varchar(40)	utf8mb4_general_ci		Nie	Brak		
6	data	date			Nie	Brak		
7	odp	tinyint(1)			Nie	0		

Rysunek 2: Tabela azyl_zapytania

3) Tabela forum_kateg

Podana tabela służy tylko do przechowywania informacji na temat kategorii jakie forum posiada. Później te informacje są wykorzystywane w mechanizmach serwisu np. po przez klucz obcy jaki jest w tabeli azyl_posty można posortować według kategorii.

#	Nazwa	Typ	Metoda porównywania napisów	Atrybuty	Null	Ustawienia domyślne	Komentarze	Dodatkowo
1	id_k 🗝️	int(40)			Nie	Brak		AUTO_INCREMENT
2	nazwa	varchar(40)	utf8mb4_general_ci		Nie	Brak		

Rysunek 3: Tabela forum_kateg

4) Tabela forum_odpowiedzi

Ta tabela odpowiada za przechowywanie informacji na temat udzielanych odpowiedzi w konkretnych zapytaniach. Jak można zauważyć tabela jest zbudowana z większości z kluczy obcych.

#	Nazwa	Typ	Metoda porównywania napisów	Atrybuty	Null	Ustawienia domyślne	Komentarze	Dodatkowo
1	id	int(50)			Nie	Brak		AUTO_INCREMENT
2	id_p	int(50)			Nie	Brak		
3	id_k	int(50)			Nie	Brak		
4	id_u	int(50)			Nie	Brak		
5	tresc	text	utf8mb4_general_ci		Nie	Brak		
6	data	date			Nie	Brak		

Rysunek 4: Tabela forum_odpowiedzi

5) Tabela news

Tabela na dzień dzisiejszy (17.03.19r) nie przechowuje żadnych istotnych danych które mechanizmy serwisu by wykorzystywały. Tabela ma w przyszłości przechowywać dane artykułów, news lub informacji które będą wyświetlane na stronie głównej (index.php).

#	Nazwa	Typ	Metoda porównywania napisów	Atrybuty	Null	Ustawienia domyślne	Komentarze	Dodatkowo
1	id	int(50)			Nie	Brak		AUTO_INCREMENT
2	tytul	tinytext	utf8mb4_general_ci		Nie	Brak		
3	tresc	text	utf8mb4_general_ci		Nie	Brak		
4	data	date			Nie	Brak		

Rysunek 5: Tabela news

6) Tabela ranga

Wybrana tabela spełnia ważną rolę w mechanizmach zabezpieczeń i pozwoleń serwisu. Każdy mechanizm wykorzystuje informacje z tej tabeli lub powiązanej poprzez relacje z innymi tabelami. Konkretna wartość która jest przypisana do rangi to ją właśnie każdy mechanizm potrzebuje do poprawnego działania według zamysłu autora.

#	Nazwa	Typ	Metoda porównywania napisów	Atrybuty	Null	Ustawienia domyślne	Komentarze	Dodatkowo
1	id_r	int(40)			Nie	Brak		AUTO_INCREMENT
2	nazwa	varchar(40)	utf8mb4_general_ci		Nie	Brak		
3	opis	varchar(100)	utf8mb4_general_ci		Nie	Brak		
4	value	int(40)			Nie	Brak		

Rysunek 6: Tabela ranga

7) Tabela user_bans

W tej tabeli są przechowywane informacje na temat banów które administracja/moderatorzy dała na forum. Dzięki informacją powiązanym można szybko sprawdzić jaki użytkownik nie ma dostępu do forum i z jakiego powodu. Też pełni ta tabela ważną rolę w przepuszczaniu użytkowników dalej do serwisu.

#	Nazwa	Typ	Metoda porównywania napisów	Atrybuty	Null	Ustawienia domyślne	Komentarze	Dodatkowo
1	id 	int(40)			Nie	Brak		AUTO_INCREMENT
2	id_u 	int(40)			Nie	Brak		
3	ban_time	int(40)			Nie	Brak		
4	powod	text	utf8mb4_general_ci		Nie	Brak		
5	data	date			Nie	Brak		

Rysunek 7: Tabela user_bans

8) Tabela user_login

Najważniejsza tabela w całym serwisie. To dzięki niej można zalogować się na forum i pytać się o dane zagadnienie. Dzięki tej tabeli można rozpoznać konkretnych użytkowników i nadawać uprawnienia ale to w innej tabeli(user_ranga).

#	Nazwa	Typ	Metoda porównywania napisów	Atrybuty	Null	Ustawienia domyślne	Komentarze	Dodatkowo
1	id 	int(10)			Nie	Brak		AUTO_INCREMENT
2	nick 	varchar(40)	utf8mb4_general_ci		Nie	Brak		
3	email	varchar(40)	utf8mb4_general_ci		Nie	Brak		
4	haslo	text	utf8mb4_general_ci		Nie	Brak		
5	data	date			Nie	Brak		

Rysunek 8: Tabela user_login

9) Tabela user_ranga

Kolejna z ważnych tabel w serwisie. To właśnie w niej są zapisane informacje kto ma jaką rangę i pozwala poprawnie rozpoznawać kto jest z administracji i komu przyzwolić na konkretne działanie.

#	Nazwa	Typ	Metoda porównywania napisów	Atrybuty	Null	Ustawienia domyślne	Komentarze	Dodatkowo
1	id 	int(30)			Nie	Brak		AUTO_INCREMENT
2	id_u 	int(40)			Nie	Brak		
3	id_r 	int(40)			Nie	Brak		

Rysunek 9: Tabela user_ranga

10) Tabela user_spend_time

Tabela która nie ma wpływu na działanie serwisu na stan (17.03.19r). W kolejnych miesiącach aktualizacji tabela będzie spełniać funkcje zbierania informacji ile dany użytkownik korzysta z forum. Dzięki tym informacją można stworzyć kolejne systemy nagradzania użytkowników za korzystanie z forum.

#	Nazwa	Typ	Metoda porównywania napisów	Atrybuty	Null	Ustawienia domyślne	Komentarze	Dodatkowo
1	id	int(40)			Nie	Brak		AUTO_INCREMENT
2	id_u	int(40)			Nie	Brak		
3	czas	int(40)			Nie	Brak		

Rysunek 10: Tabela user_spend_time

11) Tabela_time

Tabela która miała zbierać informacje o czasie korzystania konkretnych użytkowników teamspeak. Nie jest wykorzystywana w żadnych z systemów serwisu na dzień(17.03.19r).

#	Nazwa	Typ	Metoda porównywania napisów	Atrybuty	Null	Ustawienia domyślne	Komentarze	Dodatkowo
1	id	int(10)			Nie	Brak		AUTO_INCREMENT
2	nick	varchar(40)	utf8mb4_general_ci		Tak	NULL		
3	czas	int(30)			Tak	NULL		

Rysunek 11: Tabela user_time

1. OPIS UPRAWNIEŃ KONKRETNÝCH UŻYTKOWNIKÓW DO BAZY

W bazie są użytkownicy domyślni których nie zaleca się w wykorzystywaniu do dalszej eksploatacji serwisu ze względu na bezpieczeństwo. Na dzień dzisiejszy użytkownik phpmyadmin ma zmienione hasło z domyślnego i uprawnienia do korzystania z bazy danej co czyni serwis bardziej odporny na ataki ale nadal nie zaleca się korzystania z tego użytkownika. Dla bezpieczeństwa w przyszłości zostanie utworzony specjalnie dedykowany użytkownik aby uniemożliwić włamywaczowi szukanie po innych bazach.

Nazwa użytkownika	Host name	Typ	Uprawnienia	Nadawanie	Działanie
phpmyadmin	localhost	ogólny	ALL PRIVILEGES	Nie	
root	localhost	ogólny	ALL PRIVILEGES	Tak	

Ilustracja 3: Baza spis użytkowników

2. WYDAJNOŚĆ BAZY DANYCH I JEJ BEZPIECZEŃSTWO

Po wielokrotnych testach stwierdzam że baza na tą ilość informacji lub o wiele większej (parę tysięcy rekordów) jest i będzie trzymała stałą wydajność. Bezpieczeństwo oceniam na dobre ale mogło być lepsze ze względu że jeden użytkownik z którego aktualnie korzysta serwis ma dostęp do wszystkiego co stwarza zagrożenie.

III. OPIS SERWISU

1. PODSTAWOWE INFORMACJE NA TEMAT SERWISU

Serwis posiada funkcjonalności forum i w przyszłych aktualizacja możliwość do zarządzania teamspeakiem. Serwis znajduje się pod domeną Forum-azyl.pl. Serwis udostępnia wygodny panel do zarządzania całym forum. Serwis posiada dużo mechanizmów zabezpieczających co czyni ten serwis bezpiecznym i odpornym na podstawowe ataki hakerskie.

2. SPIS MECHANIZMÓW I ICH OPIS

- Mechanizm logowania się do bazy danej

Ten mechanizm dla wygody został umieszczony w osobnym pliku aby móc później go przypisywać tylko do konkretnych stron w serwisie.

```
1  <?php
2  /*****
3   * c_baza.php.php
4   * konfiguracja połączenia z bazą danych
5   *****/
6  $servername = "localhost";
7  $username = "phpmyadmin";
8  $password = " ";
9  $dbname= "azyl";
10 $conn = new mysqli($servername, $username, $password, $dbname);
11
12 if($conn->connect_error)
13 {
14     die("Błąd podczas laczania z baza danych");
15 }
16 mysqli_close($connection);
17 session_start();
18 ?>
```

Ilustracja 4: Mechanizm logowania sie do bazy

- Mechanizm logowania i rejestracji użytkownika

Zasada działania jest prosta. Z formularza dane zostają zapisane do zmiennych a dalej są wpisywane do konkretnych tabel lub przyrównane w celu porównania.

```

157 // kiedy czynności zostaną wykonane może nastąpić przekierowanie
158 //Odbierania danych z formularza
159 $nick = $_POST['inputNick'];
160 $email = $_POST['inputEmail'];
161 $data = date('Y-m-d');
162
163 //szyfrowanie hasła
164 $salt = " ";
165 $haslo = $_POST['inputPassword'] $salt;
166 $haslo = sha1($haslo);
167
168 //sprawdzenie nie działa NAPRAWIĆ
169 $sql_spr = "SELECT * FROM user_login WHERE email = '$email' || nick = '$nick'";
170 $wynik_spr = $conn->query($sql_spr);
171 if(mysqli_num_rows($wynik_spr) > 0)
172 {
173     {
174         echo "<script type='text/javascript'>window.alert('Użytkownik o podanym emailu lub nicku istnieje');</script>";
175     }
176 }
177 else
178 {
179     //dodanie nowego rekordu
180     $sql = "INSERT INTO user_login (nick, email, haslo, data)
181     VALUES ('$nick','$email','$haslo','$data')";
182     if ($conn->query($sql) == TRUE)
183     {
184         //pobranie id użytkownika nowo zarejestrowanego
185
186         $sql_id = "SELECT id FROM user_login WHERE nick = '$nick'";
187         $wynik_id = $conn->query($sql_id);
188         if($wynik_id->num_rows > 0)
189         {
190             while ($row = $wynik_id -> fetch_assoc())
191             {
192                 $id_usera = $row['id'];
193             }
194         }
195
196         //Zapytanie do bazy i szukanie id użytkownika
197         $sql_user = "SELECT * FROM ranga WHERE nazwa = 'Użytkownik'";
198         $wynik_user = $conn->query($sql_user);
199         if($wynik_user->num_rows > 0)
200         {
201             while ($row = $wynik_user -> fetch_assoc())
202             {
203                 $id_rangi = $row['id_r'];
204             }
205         }
206
207         //nadawanie rangi
208
209         $sql_ranga = "INSERT INTO user_ranga (id_u,id_r)
210         VALUES ('$id_usera','$id_rangi')";
211         $wynik_ranga = $conn->query($sql_ranga);
212     }
213     else
214     {
215         echo "Error: " . $sql . "<br>" . $conn->error;
216     }
217 }
218 $conn->close();
219 header("Location: sukces.php");
220 exit;
221

```

Ilustracja 5: Mechanizm rejestracji

```

<?php
$nick = $_POST['inputNick'];
$sql = "SELECT * FROM user_login WHERE nick = '$nick'";
$result = $conn->query($sql);

$row = $result->fetch_assoc();

if($result->num_rows == 0) {
    echo "<script type='text/javascript'>window.alert('Nie znaleziono takiego użytkownika');</script>";
}
else
{
    //echo "login ".$email."<br>hasło ".$haslo."<br>poprawne hasło ".$row["haslo"];
    //odszyfrowywanie hasła z bazy
    $salt = " ";
    $haslo = $_POST['inputPassword'].$salt;
    $haslo = sha1($haslo);
    if($haslo == $row["haslo"]) {

        //Wyszukanie i sprawdzanie dla danego użytkownika jaką ma rangę
        $id=$row['id'];
        $zapytanie_ranga = "SELECT * FROM user_ranga WHERE id_u = '$id'";
        $wynik_ranga = $conn->query($zapytanie_ranga);
        if($wynik_ranga->num_rows > 0)
        {
            while ($row = $wynik_ranga -> fetch_assoc())
            {
                $id_r=$row['id_r'];
            }
        }

        //Sprawdzenie czy użytkownik ma bana
        $zapytanie_ban = "SELECT * FROM user_bans WHERE id_u = '$id'";
        $wynik_ban = $conn->query($zapytanie_ban);
        if($wynik_ban->num_rows > 0)
        {
            while ($row = $wynik_ban -> fetch_assoc())
            {
                $ban_time=$row['ban_time'];
                $powod=$row['powod'];
            }
        }

        if ($ban_time > 0)
        {
            echo "<h3 style='color:red;'>". "Brak dostępu powód ".$powod." ban na ".$ban_time." dni."</h3>";
            exit;
        }
        else if (empty($ban_time))
        {
            echo $ban_time;
        }

        //Wyszukiwanie po id_r nazwy rangi
        $zapytanie_nazwa = "SELECT * FROM ranga WHERE id_r = '$id_r'";
        $wynik_nazwa = $conn->query($zapytanie_nazwa);
        if($wynik_ranga->num_rows > 0)
        {
            while ($row = $wynik_nazwa -> fetch_assoc())
            {
                $n_ranga=$row['nazwa'];
                $moc_pozwolen=$row['value'];
            }
        }

        //Wpisywanie wartości do sesji
        $_SESSION["nick"] = $nick;
        $_SESSION["haslo"] = $haslo;
        $_SESSION["zalogowany"] = true;
        $_SESSION["ranga"] = $n_ranga;
        $_SESSION["moc_upr"] = $moc_pozwolen;

        header("refresh:0;URL=index.php");
        echo "<script type='text/javascript'>window.alert('zalogowano');</script>";
    }
}

```

Ilustracja 6: Mechanizm logowania

-
- Mechanizm szyfrowania haseł

Ten moduł odpowiada za zaszyfrowanie hasła podanego przez użytkownika i wpisania go do bazy. Szyfrowanie odbywa metodą sha1 z tak zwaną solą czyli do każdego hasła dopisuje kombinacje cyfr i liter losowych tak aby żaden potencjalny oprawca nie mógł od razu po włamaniu do bazy odszyfrować wszystkich haseł do kont jedną metodą.

```
//szyfrowanie hasła
$salt = "XXXXXXXXXX";
$haslo = $_POST['inputPassword'].$salt;
$haslo = sha1($haslo);

//sprawdzenie nie działa NAPRAWIC
$sql_spr = "SELECT * FROM user_login WHERE email = '$email' || nick = '$nick'";
$wynik_spr = $conn->query($sql_spr);
if(mysqli_num_rows($wynik_spr) > 0)
{

```

Ilustracja 7: Mechanizm szyfrowania haseł

- Mechanizm sprawdzania poprawności haseł

Ten mechanizm działa na zasadzie przyrównania podanej wartości przez użytkownika do hasła które zostało odszyfrowane z bazy. Jeżeli hasła są poprawne mechanizm przepuści dalej a jeżeli nie wyświetli komunikat ze złe hasło.

```
140 //odszyfrowywanie hasła z bazy
141 $salt = "XXXXXXXXXX";
142 $haslo = $_POST['inputPassword'].$salt;
143 $haslo = sha1($haslo);
144 if($haslo == $row["haslo"]) {
145
146 //Wyszukanie i sprawdzanie dla danego uzytkownika jaką ma range
147 $id=$row['id'];
148 $zapytanie_ranga = "SELECT * FROM user_ranga WHERE id_u = '$id'";
149 $wynik_ranga = $conn->query($zapytanie_ranga);
150 if($wynik_ranga->num_rows > 0)
151 {
152     while ($row = $wynik_ranga -> fetch_assoc())
153     {
154         $id_r=$row['id_r'];
155     }
156 }
```

Ilustracja 8: Mechanizm sprawdzania poprawności hasła

- Mechanizm sprawdzania rangi

Polega na przyrównaniu wartości odczytanej z bazy z pomocą klauzuli where = „nick użytkownika” i sprawdzenia jaką range ma użytkownik.(Patrz wyżej ilustracja)

- Sesje

Sesja dla danego użytkownika zostaje utworzona przy logowaniu się. Do sesji zostają wpisane takie dane jak nick, moc uprawnień, czy się zalogowano, nazwę rangi. Każda z tych danych jest odpowiednio zadeklarowana. Np wartość zalogowany jest bool aby móc łatwo określić czy udostępnić forum czy nie.

```

189 //wpisywanie wartosci do sesji
190 $_SESSION["nick"] = $nick;
191 $_SESSION["haslo"] = $haslo;
192 $_SESSION["zalogowany"] = true;
193 $_SESSION["ranga"] = $n_ranga;
194 $_SESSION["moc_upr"] = $moc_pozwolen;
195
196 header("refresh:0;URL=index.php");
197 echo "<script type='text/javascript'>window.alert('zalogowano');</script>";
198 }

```

Ilustracja 9: Sesje

- Odczyt czy ma się uprawnienia do danej strony

Zasada działania tego mechanizmu jest taka że kod php odczytuje wartość rangi z sesji i sprawdza czy wartość jest odpowiednia aby uzyskać dostęp do zawartości danej strony jeśli nie wyświetli się odpowiedni komunikat.

```

96 <?php
97     if ($_SESSION["moc_upr"] > 74)
98     {
99     }
100     <br /><button href="/" onClick="op('wyswietl_1'); return false;" class="btn btn-primary button">button

```

Ilustracja 10: Mechanizm spr. uprawnień

- Odczyt czy dany użytkownik ma bana

Polega na sprawdzeniu czy użytkownik posiada bana czy nie. Jeżeli tak pojawi się informacja na temat bana, powodu i czasu trwania.

```

156 }
157 //Sprawdzenie czy uzytkownik ma bana
158 $zapytanie_ban = "SELECT * FROM user_bans WHERE id_u = '$id'";
159 $wynik_ban = $conn->query($zapytanie_ban);
160 if($wynik_ban->num_rows > 0)
161 {
162     while ($row = $wynik_ban -> fetch_assoc())
163     {
164         $ban_time=$row['ban_time'];
165         $powod=$row['powod'];
166     }
167 }
168 if ($ban_time > 0)
169 {
170     echo "<h3 style='color:red;'>". "Brak dostępu powód ". $powod. " ban na ". $ban_time. " dni."</h3>";
171     exit;
172 }
173 else if (empty($ban_time))
174 {
175     echo $ban_time;
176 }
177 //Wyszukiwanie po id_r nazwy rangi
178 $zapytanie_nazwa = "SELECT * FROM ranga WHERE id_r = '$id_r'";
179 $wynik_nazwa = $conn->query($zapytanie_nazwa);
180 if($wynik_nazwa->num_rows > 0)
181 {
182     while ($row = $wynik_nazwa -> fetch_assoc())
183     {

```

Ilustracja 11: Mechanizm spr. bana

- Mechanizm odpowiadania na pytania(formularz kontaktowy)

Ten mechanizm polega na pobraniu wartości z formularza i wpisaniu ich do bazy. Później administracja ma wypisane dane wiadomości i ma możliwość odpowiadania na nie. Przy każdej wiadomości jest zaznaczone czy zostało odpisane czy nie (patrz ilustracja).

Posty						
Mail						
<input type="text"/> Wyszukaj po id <input type="button" value="Wyszukaj"/>						
Id	Imie i nazwisko	email	Temat wiadomości	Data	Odpowiedziano	Odpowiedz
3	Aleksander Kowalski	mrhdolek14@gmail.com	test2	2019-03-01	Tak	<input type="button" value="Odpowiedz"/> <input type="button" value="Usuń"/>
2	Aleksander Kowalski	mrhdolek14@gmail.com	test	2019-03-01	Tak	<input type="button" value="Odpowiedz"/> <input type="button" value="Usuń"/>

Ilustracja 12: Mechanizm email odczyt czy zostało odpowiedziane

Później poprzez kliknięcie przycisku odpowiedz użytkownik zostaje przeniesiony na kolejny formularz i ma możliwość odpowiedzenia na pytanie. Odpowiedzi zostaje przesłana na

```

1  <?php
2  include 'c_baza.php';
3  header('Content-type: text/html; charset=utf-8');
4
5  use PHPMailer\PHPMailer\PHPMailer;
6  use PHPMailer\PHPMailer\Exception;
7
8  require 'phpmailer/src/Exception.php';
9  require 'phpmailer/src/PHPMailer.php';
10 require 'phpmailer/src/SMTP.php';
11
12 $imie = $_GET['inputAutor'];
13 $email = $_GET['nadawca_email'];
14 $tresc = $_GET['tresc'];
15 $temat = $_GET['inputTemat'];
16
17
18
19 date_default_timezone_set('Europe/Warsaw');
20
21 $mail = new PHPMailer(true);
22 try {
23     $mail->isSMTP(); // Używamy SMTP
24     $mail->Host = 'smtp.gmail.com'; // Adres serwera SMTP
25     $mail->SMTPAuth = true; // Autoryzacja (do) SMTP
26     $mail->Username = 'forum.azyl@gmail.com'; // Nazwa użytkownika
27     $mail->Password = ' '; // Hasło
28     $mail->SMTPSecure = 'tls'; // Typ szyfrowania (TLS/SSL)
29     $mail->Port = 587; // Port
30
31     $mail->CharSet = "UTF-8";
32     $mail->setLanguage('pl', '/phpmailer/language');
33
34     $mail->setFrom('forum.azyl@gmail.com', 'Forum-azyl.pl'); //nadawca
35     $mail->addAddress($email, $imie); //odbiorca
36
37
38     $mail->isHTML(true); // Format: HTML
39     $mail->Subject = $temat;
40     $mail->Body = $tresc;
41     $mail->AltBody = 'By wyświetlić wiadomość należy skorzystać z czytelnika obsługującego wiadomości w formie HTML';
42
43     $mail->send();
44     // Gdy OK:
45     $id = $_GET['id'];
46     $sql = "UPDATE azyl_zapytania SET odp = '1' WHERE id='$id'";
47     $result = $conn->query($sql);
48
49     echo "<fieldset>";
50     echo "<div id='success'>";
51     echo "<h1>Wiadomość została wysłana pomyślnie.</h1>";
52     echo "</div>";
53     echo "</fieldset>";
54     header("Refresh: 2; URL=gui_admin.php");
55
56 } catch (Exception $e) {
57     // Gdy błąd:
58     echo 'Wystąpił błąd podczas wysyłania wiadomości! Błąd: ' . $mail->ErrorInfo;
59     header("Refresh: 7; URL=gui_admin.php");
60 }
61
62
63 ?>

```

skrzynkę adresata. Ten mechanizm wykorzystuje bezpłatną bibliotekę phpmailer [link](#).

- Wyszukiwarka dla użytkowników i administracji

Ten mechanizm został podzielony ze względów estetycznych i bezpieczeństwa aby nikt z zewnątrz nie miał ingerencji w pliki administracyjne. Wyszukiwarka dla administracji różni się tym że administracja ma możliwość szukania po konkretnej kategorii i do każdej kategorii są odpowiednie pola. A użytkownicy mają wyszukiwanie po danej frazie postów z danej kategorii.

```
$dane = $_POST['wyszukaj'];

if (empty($dane))
{
    $zapytanie_kate = "SELECT * FROM azyl_posty WHERE id_k = '$id'";
}
else
{
    $zapytanie_kate = "SELECT * FROM azyl_posty WHERE (wlasiciel Like '%$dane%' AND id_k = '$id') OR (tytul_postu Like '%$dane%' AND id_k = '$id') OR (data Like '%$dane%' AND id_k = '$id')";
}
```

Ilustracja 13: Wyszukiwarka dla użytkowników

```
2 <script>
3 function pokaz_data()
4 {
5     var x = document.getElementById("poka_data");
6     var y = document.getElementById("pole_wysz");
7     x.style.display = "block";
8     y.style.display = "none";
9 }
10 function ukryj_data()
11 {
12     var x = document.getElementById("poka_data");
13     var y = document.getElementById("pole_wysz");
14     x.style.display = "none";
15     y.style.display = "block";
16 }
17 </script>
18 <form method="GET" action="gui_admin.php" name="wyszukiwarka">
19     <input type="text" name="pole_wysz" id="pole_wysz" style="display: block" />
20     <input type="date" name="poka_data" id="poka_data" style="display: none" />
21     <select name="wyb_wysz">
22         <?php
23
24
25         $zapytanie = "DESCRIBE azyl_posty";
26         $wynik = $conn->query($zapytanie);
27         if($wynik->num_rows > 0)
28         {
29             while ($row = $wynik -> fetch_assoc())
30             {
31                 if ($row['Field'] == 'data')
32                 {
33                     echo('<option value="'. $row['Field'].'" onclick="pokaz_data()">Wyszukaj po '. $row['Field']. '</option>');
34                 }
35                 else
36                 {
37                     echo('<option value="'. $row['Field'].'" onclick="ukryj_data()">Wyszukaj po '. $row['Field']. '</option>');
38                 }
39             }
40
41         }
42
43     }
44
45
46
47     ?>
48
49
50
51 </select>
52 <input type="submit" value="Wyszukaj" class="btn btn-info" />
53
54 </form>
```

Ilustracja 14: Wyszukiwarka dla administracji

- Sprawdzanie poprawności danych

Poprawność jest sprawdzana na poziomie html (bootstrap) czyli każdy input ma dokładnie zadeklarowaną jaki typ ma

przyjąć a jaki nie. Jeżeli użytkownik się uprze że chce wstawić inny typ zmiennej poprzez edycje strony www przez przeglądarkę to jest to zablokowane z poziomu php.

```
<div class="container">

  <div class="align-items-center">
    <form class="form-signin" method="post" id="logowanie" action="/p_logowanie.php">
      <h2 class="form-signin-heading" style="text-align: center;">Logowanie</h2>
      <label for="inputEmail" class="sr-only">Nick</label>
      <input type="text" id="inputNick" name="inputNick" class="form-control" placeholder="Nick" required="" autofocus="" maxlength="40">
      <label for="inputPassword" class="sr-only">Hasło</label>
      <input type="password" name="inputPassword" id="inputPassword" class="form-control" placeholder="Hasło" required="">
      <div class="checkbox">
        <input type="checkbox" value="remember-me"> Zapamiętaj mnie
      </div>
      <button class="btn btn-lg btn-primary btn-block" type="submit">Zaloguj</button>
    </form>
  </div>
</div>
```

Ilustracja 15: Panel logowania

- Zapis/odczyt pliku

W serwisie istnieje ta funkcja jest ona zaszyta przy edytowaniu regulaminu forum.

```
638 <?php
639 //edycja regulaminu w panelu cruda
640 if(isset($_POST['zapisz_regulamin']))
641 {
642     $regulamin = $_POST['edycja_regulaminu'];
643
644     /// otwarcie pliku do zapisu
645     $fp = fopen("/var/www/html/regulamin.txt", "w+") or die("Nie można otworzyć pliku");
646     fwrite($fp, $regulamin);
647     fclose($fp);
648     echo "<script type='text/javascript'>window.alert('Udało się zapisać');</script>";
649 }
650 }
651 }
652 }
653 }
```

Ilustracja 16: Zapis/odczyt pliku

- Cookies

Ciasteczka w serwisie są wykorzystywane do zapamiętywania czy dany użytkownik już był w serwisie. Dana informacja o wykorzystywaniu cookies jest wyświetlana każdemu nowemu użytkownikowi.

```
1 function WHCheckCookies(){
2     if(localStorage.cookies_accepted) {
3         var cookies_message = document.getElementById("cookies-message");
4         cookies_message.style.display="block"
5     }
6 }
7 function WHCloseCookieswindow(){
8     localStorage.cookies_accepted = true;
9     document.getElementById("cookies-message-container").removeChild(document.getElementById("cookies-message"));
10 }
11
12 window.onload = WHCheckCookies;
```

Ilustracja 17: cookies

IV. ZARZĄDZANIE UŻYTKOWNIKAMI NA FORUM

1. PANEL ADMINA

Administracja jest zdolna do wyedytowania każdej wartości z bazy poprzez odpowiednie formularze które są adekwatnie zabezpieczone(18.03.19r).Každy z formularzy jest responsywnych czyli reaguje na zmiany w bazie danych.Jest część rzeczy gdzie administracja może tylko wykonywać.Każda grupa moderatorów,administratorów czy użytkowników ma adekwatne prawa.Administrator może nadawać różne rangi za pomocą edycji użytkownika i także usuwać użytkowników razem z wszystkimi postami ich.

Panel admina znajduje się pod plikiem: **gui_admin.php**

```
159
160
161 $wynik = $conn->query($zapytanie);
162 if($wynik->num_rows > 0)
163 {
164     while ($row = $wynik -> fetch_assoc())
165     {
166         echo '<tr>';
167         echo '<td>'.$row['id'].'</a></td>';
168         echo '<td>'.$row['tytul_postu'].'</td>';
169         echo '<td>'.$row['wlasiciel'].'</td>';
170         echo '<td>'.$row['tresc'].'</td>';
171         echo '<td>'.$row['data'].'</td>';
172         echo '<td>'.<a href = "edycja_tabelek_posty.php?id='.$row["id"].'" class="btn btn-info"> Edycja </a>'.<a href
173             = "usuwanie_tabelek_posty.php?id='.$row["id"].'" class="btn btn-danger"> Usuń </a>'.</td>';
174     }
175     echo '</tr>';
176 }
```

Ilustracja 18: Kawałek kodu panelu admina

2. PANEL UŻYTKOWNIKA

Użytkownik na forum może dodawać posty,komentować i usuwać je jeżeli jest ich właścicielem.Ma także możliwość edycji własnych danych poprzez odpowiedni panel.

```

96
97 <h3 class="text-muted">Twoje dane:</h3>
98 <?php
99
100 $nick = $_SESSION['nick'];
101
102 $zapytanie = "SELECT * FROM user_login WHERE nick='".$nick."'";
103
104 $wynik = $conn->query($zapytanie);
105
106 if($wynik->num_rows > 0)
107 {
108     while($row = $wynik -> fetch_assoc())
109     {
110         echo '
111         <form method="GET" id="Edycja" action="/user_profil_run.php">
112         <table>
113             <tr><td>Nick : </td><td><input type="text" id="inputNick" name="inputNick" maxlength="50" value="'.$row["nick"].'
114             "></td></tr>
115             <tr><td>Email : </td><td><input type="text" id="inputEmail" name="inputEmail" maxlength="50" value="'.$row["email"].'
116             "' type="email"></td></tr>
117             <tr><td>Hasło : </td><td><input type="password" id="inputPassword" name="inputPassword" maxlength="50" value="'.$row
118             ["haslo"].' "></td></tr>
119         </table>
120         <div style="text-align: left;">
121         <button type="submit" name="edycja" class="btn btn-sm btn-success">Edycja</button>
122         <input type="reset" value="Reset" class="btn btn-sm btn-danger">
123         </div>
124         </form>;
125         echo '</table>';
126     }
127     else{
128         echo "Brak danych";
129     }
130 }
131 ?>
132
133
134
135
136
137 <!-- Site footer -->
138 <footer class="footer">
139     <p>© 2019 Aleksander Kowalski 'MrHDOLEK'</p>
140 </footer>
141
142 </div> <!-- /container -->
143 </body>
144 </html>

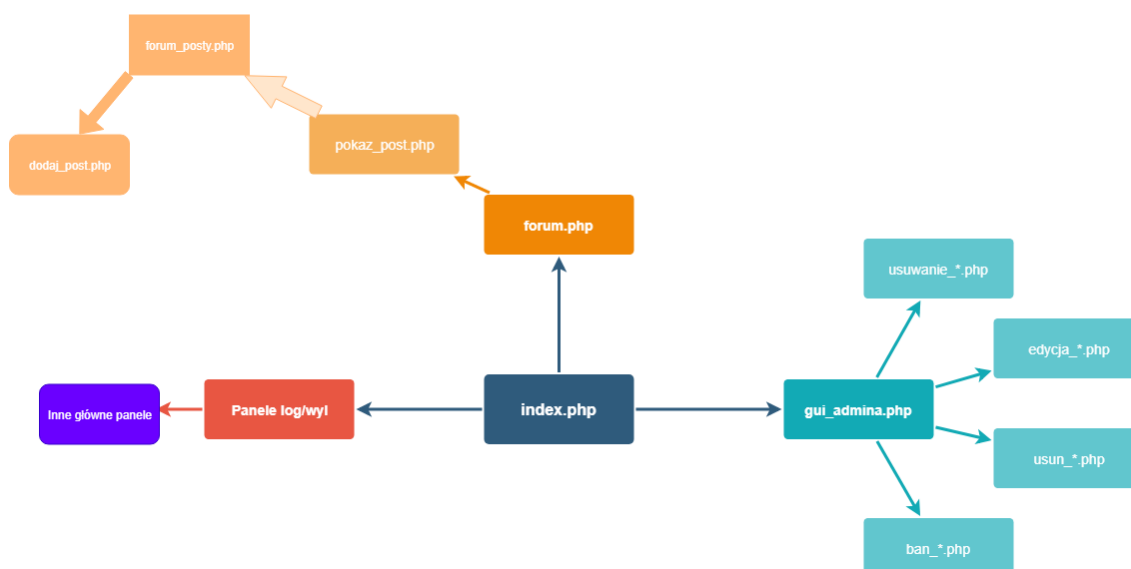
```

Ilustracja 19: Panel użytkownika

V. SPIS PLIKÓW I ICH WŁAŚCIWOŚCI

1. CO GDZIE JEST ?

Każdy plik jest za tytułowany adekwatnie za co odpowiada.



Rysunek 12: Schemat plików

2. SPIS UPRAWNIEŃ POTRZEBNYCH OD ODCZYTU PLIKU

W tabeli jest pokazane jaka ranga nie ma dostępu do jakich plików.

Ranga	Pliki	Uprawnienia
Użytkownik	Wszystkie od forum i panel użytkownika	Odczyt/zapis
Moderator	Wszystkie (prócz wybranych sekcji kodu od gui_admina)	Odczyt/zapis
Administrator	Wszystkie	Odczyt/zapis/wykonanie

VI. PODSUMOWANIE

Poprzez wykonanie projektu uświadomiłem sobie jaki jest mój stan wiedzy i w jakich technologiach jestem biegły. Dzięki temu wiem w którym kierunku chce się rozwijać i jakiej wiedzy poszukiwać. Ten serwis dla mnie jest jeszcze nie skończony brakuje paru funkcji które chce wprowadzić wraz aktualizacjami serwisu. Także ten projekt mi pokazał jak ważne jest pisanie komentarzy i klarownego kodu czyli trzymania zasad wzorców projektów dzięki którym każdy programista który je zna będzie wiedział jak serwis jest zbudowany od zaplecza kodowego i umożliwi w przyszłości łatwiejsze aktualizowanie danej aplikacji, serwisu lub witryny. Nauczyłem się wiele przydatnych sztuczek i paru dobrych bibliotek jak phpmailer czy nawet przypomnienie bootstrapa w ver. 3.3.1. Wiem że serwis jest oparty na przestarzałym standardzie ale najszybciej i najpewniej się poruszałem w nim. Na dzień dzisiejszy jest ver.4.1 (18.03.19r).

Literatura

1. php-i-mysql-dla-kazdego-wydanie-iii-marcin-lis-Ebookpoint.pl.pdf Marcin Lis
2. php obiekty, wzorce i narzędzia wyd V Matt Zandstra
3. Ciepłe słowo J. Miluta :)
4. Forum pasja informatyki
5. GITHUB
6. phpinfo()

Spis ilustracji

Ilustracja 1: Schemat bazy danych.....	6
Ilustracja 2: Schemat relacji.....	7
Ilustracja 3: Baza spis użytkowników.....	11
Ilustracja 4: Mechanizm logowania się do bazy.....	12
Ilustracja 5: Mechanizm rejestracji.....	14
Ilustracja 6: Mechanizm logowania.....	15
Ilustracja 7: Mechanizm szyfrowania haseł.....	16
Ilustracja 8: Mechanizm sprawdzania poprawności hasła.....	16
Ilustracja 9: Sesje.....	17
Ilustracja 10: Mechanizm spr. uprawnień.....	17
Ilustracja 11: Mechanizm spr. bana.....	17
Ilustracja 12: Mechanizm email odczyt czy zostało odpowiedziane.....	18
Ilustracja 13: Wyszukiwarka dla użytkowników.....	19
Ilustracja 14: Wyszukiwarka dla administracji.....	19
Ilustracja 15: Panel logowania.....	20
Ilustracja 16: Zapis/odczyt pliku.....	20
Ilustracja 17: cookies.....	20
Ilustracja 18: Kawałek kodu panelu admina.....	21
Ilustracja 19: Panel użytkownika.....	22

Spis Rysunków

Rysunek 1: Tabela azyl_posty.....	8
-----------------------------------	---

Rysunek 2: Tabela azyl_zapytania.....	8
Rysunek 3: Tabela forum_kateg.....	8
Rysunek 4: Tabela forum_odpowiedzi.....	9
Rysunek 5: Tabela news.....	9
Rysunek 6: Tabela ranga.....	9
Rysunek 7: Tabela user_bans.....	10
Rysunek 8: Tabela user_login.....	10
Rysunek 9: Tabela user_ranga.....	10
Rysunek 10: Tabela user_spend_time.....	11
Rysunek 11: Tabela user_time.....	11
Rysunek 12: Schemat plików.....	22