Zastosowawanie Programowania Obiektowego Informatyka III Semestr

Piłka nożna - Katalog (Aplikacja PHP)

Jakub Achtelik, Oliwier Budnik



Politechnika Koszalińska Koszalin 2023

Spis Treści

1	Kon	cepcja i funkcjonalność aplikacji	2
	1.1		2
		1.1.1 Środowisko uruchomieniowe:	2
	1.2	Założenia przyjęte dla tworzenia aplikacji	3
		,	3
			3
	1.3		3
	1.4	Wykaz funkcjonalności aplikacji	4
	1.5	Prototyp	5
_	Die		_
2			7
		7 (
	2.2	Opis	8
3	Diac	gram klas	9
	3.1		9
	3.2	Opis przeznaczenia klas	9
	3.3	Models	9
		3.3.1 Aplikacja	9
		3.3.2 BazaDanych	9
		3.3.3 ZapytaniaSql	9
		3.3.4 FiltrowanieSql	9
		3.3.5 OperacjePilkarzy	9
		3.3.6 WyswietlaniePilkarzy	9
		3.3.7 PobieraczObrazowWikipedia	9
	3.4	Controllers	9
		3.4.1 Autoryzacja	9
		3.4.2 FiltrowanieKontroler	9
		3.4.3 KontrolerDanych	9
		3.4.4 PilkarzPost	9
		3.4.5 Autoryzacja	9
			9
	3.5		9
			9
		3.5.2 SzablonHtml	
	3.6	Helpers	
		3.6.1 BazaDanychHelper	0
		3.6.2 FormularzHelper	
	3.7	Projekt	
	3.8	FileLoader	0
4	Droi	ekt Bazy Danych 1	1
1	4.1	Model bazy danych - relacyjna SQL	
		Opis	
		4.2.1 Ustawienie uprawnień dla użytkownika bazy danych w SQL	
	4.3	Rysunek	

Piłka nożna - Katalog (Aplikacja PHP)

5	Opis działania i obsługi aplikacji	12			
	Opis działania i obsługi aplikacji 5.1 Przewodnik	12			
	5.2 Zrzuty ekranu	12			
6	Wnioski 1				
	6.1 Podsumowanie				
	6.2 Podział pracy	13			
7	Bibliografia - źródła	14			

1 Koncepcja i funkcjonalność aplikacji

1.1 Założenia przyjęte dla tworzenia aplikacji:

Rodzaj aplikacja: Aplikacja internetowa (architektura klient-serwer) Aplikacja uruchamina lokalnie na komputerach oraz zdalnie na serwerze VPS. Aplikacja jest dososowana również do urządzeń moblinych.

Środkowisko lokalne: Windows 10, Ubuntu 22.04 LTS

Środowisko produkcyjne (serwer): VPS Linux (Ubuntu 20.04 LTS):

1.1.1 Środowisko uruchomieniowe:

Dla Windows:

• XAMPP 8.2.4 (serwer HTTP Apache, Serwer bazy danych MariaDB, interpreter PHP)

Dla Ubuntu:

- PHP 8.1.2-1ubuntu2.14 Development Server (serwer HTTP + interpreter PHP)
- mysql-server ver 8.0.34-0ubuntu0.22.04.1 (serwer bazy danych MySQL)

Dla VPS:

- Serwer HTTP Apache2
- Serwer MySQL MariaDB
- Interpreter PHP 8+ ...

1.2 Założenia przyjęte dla tworzenia aplikacji

1.2.1 Cechy

- 1. Zastosowanie statycznego typowania (zmiennych, funkcji, metod, pól klasy), podobnie jak w językach C/C++, Java, C#. Jest to bardziej przewidywalene i pozwala narzucić określony typ np. zwracanej zmiennej, aby uniknać wielu błedów. Domyślnie PHP nie wymaga statycznego typowania.
- 2. Podział projektu na wiele plików według struktury MVC Model-View-Controller (pol. Model-Widok-Kontroler):
 - Model jest pewną reprezentacją problemu bądź logiki aplikacji.
 - Widok opisuje, jak wyświetlić pewną część modelu w ramach interfejsu użytkownika.
 - · Kontroler przyjmuje dane wejściowe od użytkownika i reaguje na jego poczynania
- 3. Logika aplikacji będzie zawarta w sposób obiektowy w klasach, każda klasa to osobny plik.

1.2.2 Ograniczenia:

- PHP jest podatny na pewne rodzaje ataków, takich jak na przykład wstrzykiwanie SQL, dlatego bezpieczństwo
 aplikacji nie jest na najwyższym możliwym poziomie i szczegłówa konfiguracja zabezpieczeń nie jest łatwa do
 wdrożenia w krótkim czasie
- PHP jest językiem interpretowanym dlatego wydajność w stosunku do języków komplilowanych jest niższa
- PHP nie posiada wszystkich elementów obiektowowych znanych z innych języków
- Ograniczony czas, przez co nie można zawrzeć wszystkich celów w wzorcowy sposób zgodny w 100% z dokumentacją
- Ograniczenie aktualnej wiedzy, przez co niektóre elementy projektu mogą stanowić wyzwanie

1.3 Narzędzia programistyczne:

Język: PHP 8+ OOP

Dodatkowe biblioteki: mysqli (łączenie się z bazą danych)

Dodatkowe technologie: HTML,CSS, JavaScript, MySQL, FontAwesone(ikonki)

IDE: Visual Studio Code + PHP Code Extenions

GIT - System Kontroli Wersji

Figma – Prototypowanie wyglądu aplikacji

Trello – zarządzanie zadaniami w zespole

Przeglądarka internetowa – Posiadająca narzędzia Chrome DevTools

Pakiet make – automatyzacja poleceń w terminalu

FileZilla - klient FTP

LaTeX - skład tekstu do sprawozdania

PlantUML - narzędzie do tworzenia rysunków i schematów

1.4 Wykaz funkcjonalności aplikacji

Interfejs webowy, zarządzanie bazą danych z poziomu przeglądarki internetowej:

- edycje, usuwanie, dodawanie nowego piłkarza,
- · sortowanie oraz wyświetalnie zdjęć,
- · wyszukiwanie po nazwisku, imieniu itp.
- filtrowanie szczegłówe po np. kraju, pozycji itp.
- · logowanie oraz autoryzacja użytkownika przeglądającego aplikacje
- dodawanie/edycje, zdjęcia piłkarza

Użytkownik może za pomocą przegląrki internetowej:

- · połączyć się z serwerem na którym hostowana jest aplikacja
- zalogować się do panelu poprzez formularz logowania, uzyskać autoryzacje
- Panel umożliwia przeglądanie katalogu piłkarzy w przystępnej formie oraz inne operacje (edycja, usuwaniem, filtrowanie itp.).
- Użytkownik końcowy (klient) nie musi posiadać znajomości obsługi relacyjnej bazy danych aby w intuakcyjny sposób zarządzać aplikacją.

1.5 Prototyp

Prototyp graficzny interfejsu użytkownika. Wykonane w programie Figma.



Rysunek 1: Widok strony głównej



Rysunek 2: Widok ekranu logowania



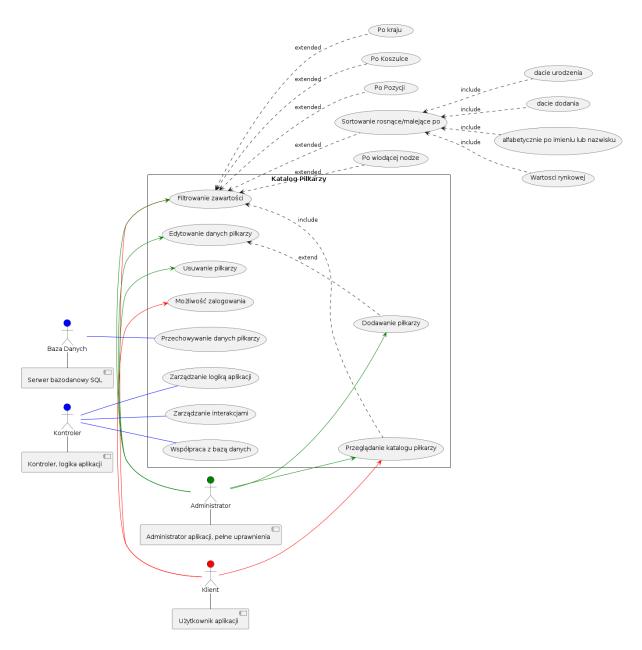
Rysunek 3: Widok strony głównej



Rysunek 4: Widok strony głównej

2 Diagram przypadków użycia

2.1 Rysunek (diagram)



Rysunek 5: Diagram przypadków użycia

2.2 Opis

Aktorami osobowymi są: Administrator oraz Klient, są to odbiorcy aplikacji **Aktorami bezosobowymi**, abstrakcyjnymi są:

- Kontroler reprezentuje logikę aplikacji
- BazaDanych reprezentuje miejsce, gdzie informację so przechowywane

Podstawowe operacje aktorów:

- Klient -> (Przeglądanie katalogu piłkarzy)
- Klient -> (Filtrowanie zawartości)
- Klient -> (Możliwość zalogowania)
- Administrator -> (Przeglądanie katalogu piłkarzy)
- Administrator -> (Filtrowanie zawartości)
- Administrator -> (Dodawanie piłkarzy)
- Administrator -> (Edytowanie danych piłkarzy)
- Administrator -> (Usuwanie piłkarzy)
- BazaDanych (Przechowywanie danych piłkarzy)
- Kontroler (Zarządzanie interakcjami)
- Kontroler (Współpraca z bazą danych)
- Kontroler (Zarządzanie logiką aplikacji)

3 Diagram klas

- 3.1 Rysunek (diagram)
- 3.2 Opis przeznaczenia klas
- 3.3 Models
- 3.3.1 Aplikacja
- 3.3.2 BazaDanych
- 3.3.3 ZapytaniaSql
- 3.3.4 FiltrowanieSql
- 3.3.5 OperacjePilkarzy
- 3.3.6 WyswietlaniePilkarzy
- 3.3.7 PobieraczObrazowWikipedia

Klasa za pomocą otwartego API Wikipedia, pobiera adres URL do głównego zdjęcia piłkarza z artykułu na Wikipedii. Zdjęcie jest pobierane podczas dodawania nowego piłkarza bądź jego edycji i zapisywana w bazie danych w taleli awatar"w postaci linku.

3.4 Controllers

- 3.4.1 Autoryzacja
- 3.4.2 FiltrowanieKontroler
- 3.4.3 KontrolerDanych

Ta klasa służy do odbierania danych od użytkownika z poziumu aplikacji, używając metod GET (parametry z linku) oraz POST (niejawnych, przesyłanych z formularzy.)

- 3.4.4 PilkarzPost
- 3.4.5 Autoryzacja
- 3.4.6 ZarzadzaniePilkarzam
- 3.5 Views
- 3.5.1 StronaHtml

Zawiera szkielet strony HTML, taki jak sekcja <head> <body> <footer> czy <header>. Dzięki temu rozwiązaniu można ustawić Podstawowe dane takie jak tytuł strony, autorów, w kiklu miejscach na stronie, wczytywana z pliku konfiguracyjnego **KonfiguracjaApp.php**

- 3.5.2 SzablonHtml
- 3.6 Helpers
- 3.6.1 BazaDanychHelper
- 3.6.2 FormularzHelper
- 3.7 Projekt

```
<?php
   // klasa do uruchamnia projektu
 3
4
   namespace Pilkanozna;
5
6
  use Pilkanozna\Controller\KontrolerStrony;
7
   use Pilkanozna\Views\StronaHtml;
8
9
10 final class Projekt
11
   {
12
       public static function Uruchom(): void
13
14
            $Aplikacja = new KontrolerStrony;
15
            $Strona = new StronaHtml;
16
17
18
            $Strona->Head();
            $Strona->Header();
19
20
            $Aplikacja->Routing();
            $Strona->Footer();
21
22
       }
23
24
25 }
```

3.8 FileLoader

4 Projekt Bazy Danych

4.1 Model bazy danych - relacyjna SQL

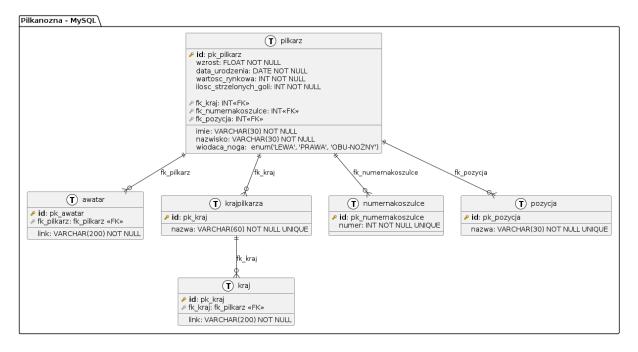
Baza danych jest relacyjna oraz obsługiwana jest na serwerze SQL.

4.2 Opis

4.2.1 Ustawienie uprawnień dla użytkownika bazy danych w SQL

```
1 CREATE USER 'projekt'@'localhost' IDENTIFIED BY 'Pracownia107!';
2 GRANT ALL PRIVILEGES ON pilkanozna.* TO 'projekt'@'localhost';
```

4.3 Rysunek



Rysunek 6: Diagram tabel bazy danych pilkanozna

- 5 Opis działania i obsługi aplikacji
- 5.1 Przewodnik
- 5.2 Zrzuty ekranu

- 6 Wnioski
- 6.1 Podsumowanie
- 6.2 Podział pracy

7 Bibliografia - źródła

Literatura

- [1] php.net Oficjalna Dokumentacja PHP https://www.php.net/docs.php
- [2] Medium Informacje na temat MVC w PHP https://medium.com/@iamjoestack/how-to-build-a-custom-php-mvc-framework-e5a23da8f73d
- [3] Konfiguracja Serwera SQL Ubuntu VPS https://ubuntu.com/server/docs/databases-mysql
- [4] Konfiguracja Serwera HTTP Apache 2 https://httpd.apache.org/docs/2.4/
- [5] Tworzenie diagramów i schematów PlantUML https://plantuml.com/
- [6] Pobieranie obrazów z API Wikipedia https://stackoverflow.com/questions/8363531/accessing-main-picture-of-wikipedia-page-by-api