

Politechnika Poznańska

Wydział Informatyki

Dawid Wiktorski

Projekt – CMS

Poznań, 2019

Spis treści

1.	Opis zadania	3
2.	Architektura.....	3
2.1.	Aktorzy systemu	3
2.2.	Wymagania funkcjonalne.....	3
2.3.	Wymagania pozafunkcjonalne	4
2.4.	Diagramy przypadków użycia	4
2.5.	Projekt bazy danych	6
3.	Technologie informatyczne	10
3.1.	Python, Django	10
3.2.	SQLite	10
3.3.	HTML, CSS, Bootstrap, TinyMCE.....	10
3.4.	System kontroli wersji	10
3.5.	Google Analytics	10
4.	Opis modułów.	11
5.	Instrukcja obsługi.....	11
6.	Instrukcja instalacji	11
7.	Testy aplikacji	11
8.	Repozytorium	12

1. Opis zadania

Zadanie obejmuje utworzenie systemu zarządzania treścią (CMS).

2. Architektura

W rozdziale znajduje się opis architektury systemu.

2.1. Aktorzy systemu

W systemie wyróżnia się takich aktorów jak:

- użytkownik niezalogowany – jest to aktor, który nie jest zalogowany w systemie,
- użytkownik zalogowany – jest to aktor, który poprawnie zalogował się w systemie,
- aplikacja internetowa – jest to aplikacja połączona z bazą danych, która odpowiada za funkcjonowanie systemu.

2.2. Wymagania funkcjonalne

Funkcje zostały pogrupowane w tabelach ze względu na typ aktora. Tabela dla aktora „aplikacja internetowa” nie została utworzona, ponieważ każda wymieniona funkcjonalność dotyczy tegoż aktora. W tabeli 1 przedstawiono funkcje dla użytkownika niezalogowanego.

Tabela 1. Wymagania funkcjonalne dla użytkownika niezalogowanego

Lp.	Funkcja	Opis
1.	Logowanie	W celu zalogowania się do systemu, należy podać login oraz hasło.
2.	Wyświetlanie najnowszych wpisów	Wyświetlanie 9 najświeższych wpisów na stronie głównej bloga.
3.	Wyświetlanie wszystkich artykułów	Możliwość wyświetlenia wszystkich artykułów wraz z jego tytułem oraz zdjęciem.
4.	Wyświetlanie wszystkich testów	Możliwość wyświetlenia wszystkich wpisów z kategorii „testy”.
5.	Wyświetlanie danych kontaktowych	Wyświetlenie m. in. numeru telefonu i adresu e-mail.
6.	Wysłanie wiadomości	Za pomocą odpowiedniego formularza użytkownik ma możliwość skontaktowania się z administracją bloga.
7.	Wyświetlenie opisu autora bloga	Wyświetlenie strony „O mnie”.
8.	Wyświetlenie galerii zdjęć	Wyświetlenie wszystkich zdjęć.
9.	Wyszukiwanie wpisów po tytule	Możliwość wyszukiwania artykułów i testów po nazwie.

Użytkownik zalogowany ma możliwość korzystania z funkcji użytkownika niezalogowanego od numeru 2 do 9, które przedstawiono w tabeli 1. W tabeli 2 ukazano funkcjonalności dla użytkownika zalogowanego.

Tabela 2. Wymagania funkcjonalne dla użytkownika zalogowanego.

Lp.	Funkcja	Opis
1.	Wylogowanie	Wylogowanie użytkownika.
2.	Zmiana tekstu oraz stylu w stopce	Możliwość zmiany treści oraz wyglądu stopki.
3.	Dodanie artykułu	Dodanie wpisu w kategorii „artykuły”.
4.	Dodanie testu	Dodanie wpisu w kategorii „testy”.
5.	Edycja wpisu	Możliwość edytowania wpisu: zmiana koloru tła, tytułu, zdjęcia, wyświetlenia na stronie głównej, ukrycia wpisu, zmiana kategorii, zmiana daty publikacji, autora oraz treści wpisu.
6.	Zmiana numeru telefonu	Możliwość zmiany telefonu oraz stylu wyświetlania.
7.	Zmiana adresu e-mail	Możliwość zmiany adresu e-mail oraz stylu wyświetlania.
8.	Zmiana tekstu na stronie „O mnie”	Możliwość zmiany tekstu na stronie „O mnie”.
9.	Zmiana głównego zdjęcia oraz tła strony	Zmiana głównego zdjęcia i tła strony.
10.	Przeglądanie logowań na stronie	Możliwość przeglądania kto, kiedy oraz czy się powiodło logowanie.
11.	Przeglądanie wysłanych wiadomości	Przeglądanie wysłanych wiadomości od innych użytkowników.
12.	Edycja „Navbar”	Zmiana wyświetlonych tekstów, koloru tekstu oraz tła.
13.	Edycja przycisków	Zmiana koloru wszystkich przycisków.

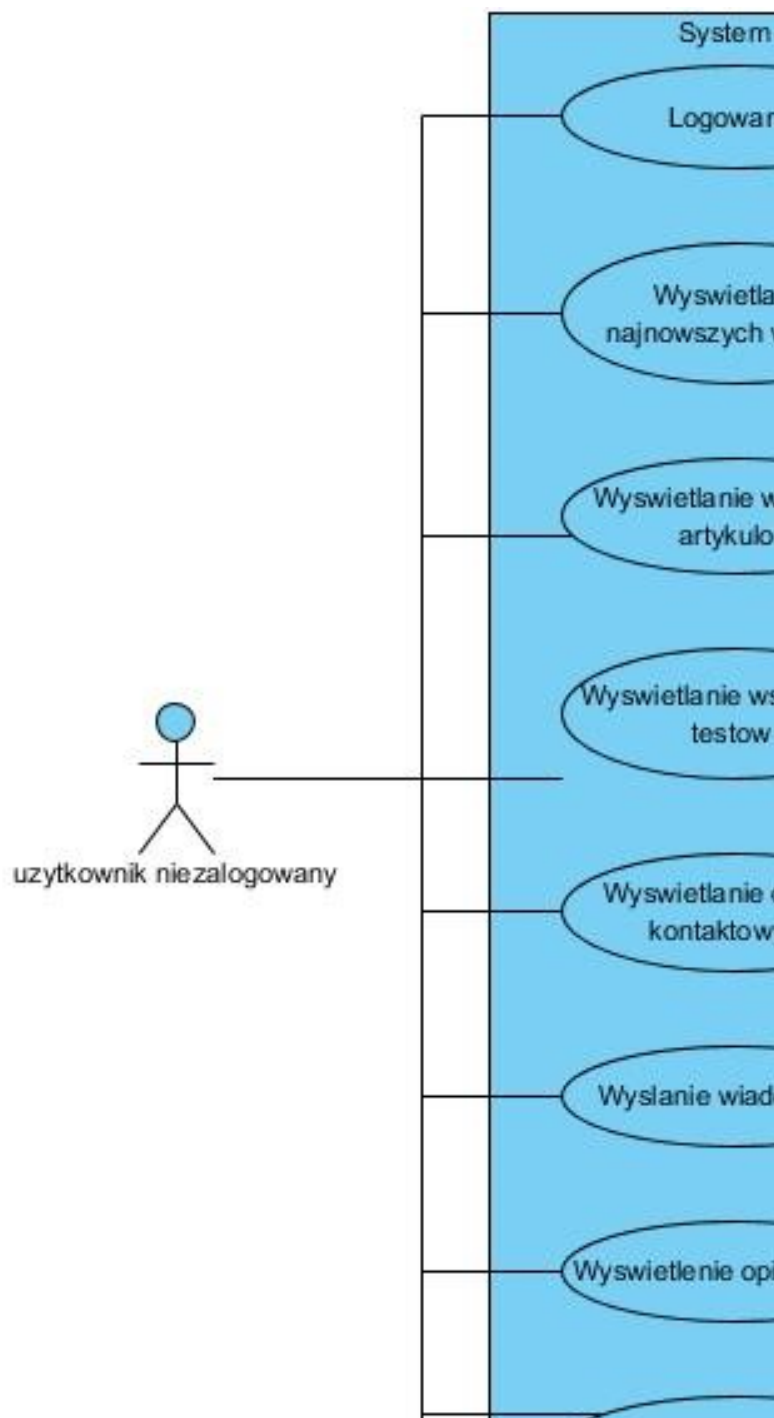
2.3. Wymagania pozafunkcjonalne

W systemie założono takie wymagania pozafunkcjonalne jak:

- angielski język interfejsu strony internetowej,
- działanie na współczesnych przeglądarkach internetowych,
- wykorzystanie technologii Django.

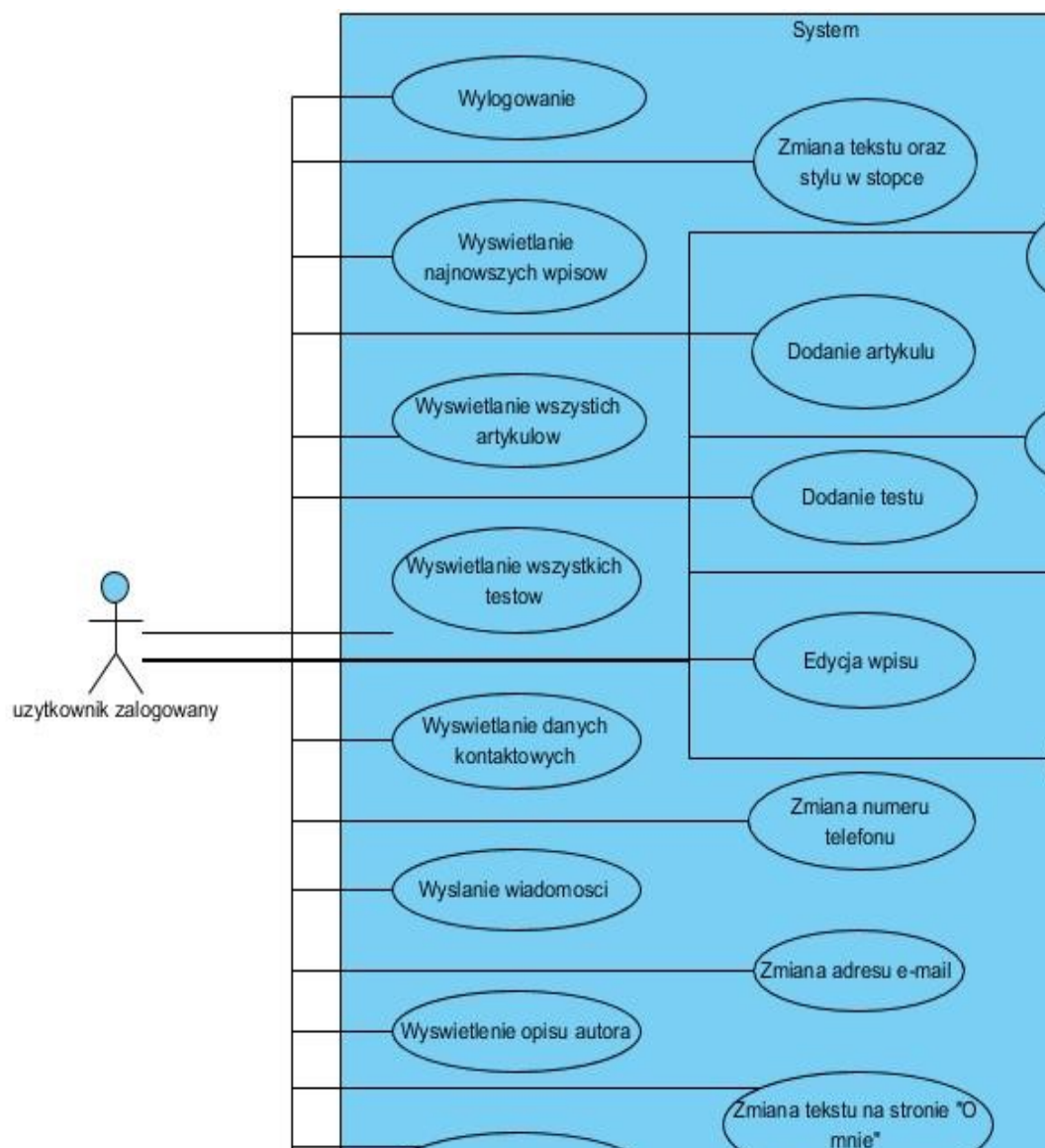
2.4. Diagramy przypadków użycia

W celu dokładniejszego zobrazowania systemu, utworzono osobne diagramy przypadków użycia dla każdego aktora. Na rysunku 1. przedstawiono diagram przypadków użycia dla użytkownika niezalogowanego.



Rysunek 1 Diagram przypadków użycia dla użytkownika niezalogowanego

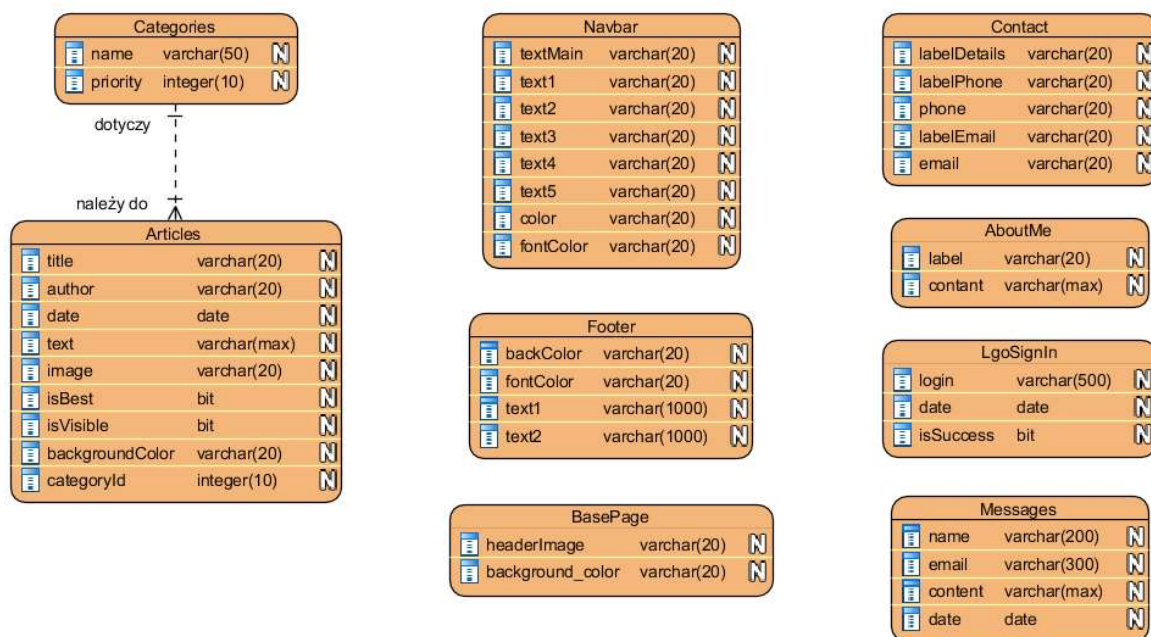
Na rysunku 2. Przedstawiono diagram przypadków użycia dla użytkownika zalogowanego.



Rysunek 2 Diagram przypadków użycia dla użytkownika zalogowanego

2.5. Projekt bazy danych

W tym podrozdziale pokazano schemat relacyjny bazy danych (rysunek 3.) w raz z opisem poszczególnych tabel.



Rysunek 3 Schemat relacyjny bazy danych

W tabeli 3 opisano tabelę Categories składającą się z 2 kolumn. W tabeli przechowywane są kategorie wpisów.

Tabela 3. Opis tabeli Categories

Nazwa kolumny	Opis
name	Nazwa kategorii.
priority	Priorytet, wskazuje jaka będzie kolejność kategorii.

W tabeli 4 opisano tabelę Articles składającą się z 9 kolumn. W tabeli przechowywane są wpisy.

Tabela 4. Opis tabeli Articles

Nazwa kolumny	Opis
title	Tytuł wpisu.
author	Autor wpisu.
date	Data wpisu.
text	Tekst wpisu wraz ze stylem oraz dodanymi zdjęciami
image	Zdjęcie wpisu.
IsBest	Wartość wskazuje, czy wpis ma być pokazany na stronie głównej.
isVisible	Wartość wskazuje, czy wpis ma być widoczny dla użytkowników.
backgroundColor	Kolor tła artykułu.
categoryId	Klucz obcy z tabeli Categories.

W tabeli 5 opisano tabelę Navbar składającą się z 8 kolumn. W tabeli przechowywane są dane dotyczące elementu Navbar.

Tabela 5. Opis tabeli Navbar

Nazwa kolumny	Opis
textMain	Treść napisu po lewej stronie Navbar.
text1	Napis na pierwszym przycisku Navbar.
text2	Napis na drugim przycisku Navbar.
text3	Napis na trzecim przycisku Navbar.
text4	Napis na czwartym przycisku Navbar.
text5	Napis na piątym przycisku Navbar.
color	Kolor tła Navbar.
fontColor	Kolor czcionki w Navbar.

W tabeli 6 opisano tabelę Footer składającą się z 4 kolumn. W tabeli przechowywane są dane dotyczące stopki.

Tabela 6. Opis tabeli Footer

Nazwa kolumny	Opis
backColor	Kolor tła stopki.
fontColor	Kolor napisu znajdującego się na samym dole stopki.
text1	Tekst stopki znajdujący się z lewej strony.
text2	Tekst stopki znajdujący się z prawej strony.

W tabeli 7 opisano tabelę BasePage składającą się z 2 kolumn. W tabeli przechowywane są informacje o szablonie strony.

Tabela 7. Opis tabeli BasePage

Nazwa kolumny	Opis
headerImage	Ścieżka do zdjęcia znajdującego się na samej górze strony.
background_color	Kolor tła bloga.

W tabeli 8 opisano tabelę Contact składającą się z 5 kolumn. W tabeli przechowywane są informacje umieszczone na stronie z kontaktem.

Tabela 8. Opis tabeli Contact

Nazwa kolumny	Opis
labelDetails	Styl i napis pierwszej etykiety.
labelPhone	Styl i napis etykiety dotyczącej telefonu.
phone	Styl i napis telefonu.
labelEmail	Styl i napis etykiety dotyczącej adresu e-mail.
email	Styl i napis adresu e-mail

W tabeli 9 opisano tabelę AboutMe składającą się z 2 kolumn. W tabeli przechowywane są dane związane ze stroną „o mnie”.

Tabela 9. Opis tabeli AboutMe

Nazwa kolumny	Opis
label	Styl i napis etykiety.
content	Styl i tekst.

W tabeli 10 opisano tabelę LogSignIn składającą się z 3 kolumn. W tabeli przechowywane są dane logowania.

Tabela 10. Opis tabeli LogSignIn

Nazwa kolumny	Opis
login	Wpisany login podczas logowania.
date	Data logowania.
isSuccess	Informacja czy logowanie się powiodło.

W tabeli 11 opisano tabelę Messages składającą się z 4 kolumn. W tabeli przechowywane są wiadomości wysłane od użytkowników.

Tabela 11. Opis tabeli Messages

Nazwa kolumny	Opis
name	Wpisana nazwa użytkownika.
Email	Wpisany adres e-mail użytkownika.
content	Treść wiadomości.
Date	Data utworzenia wiadomości.

3. Technologie informatyczne

Tworząc CMS wykorzystano technologie webowe. W tym rozdziale zostały opisane wykorzystywane technologie.

3.1. Python, Django

Python razem z Django umożliwia utworzenie aplikacji w krótkim czasie. Technologie wyróżniają się dużą stabilnością działania.

3.2. SQLite

Dla małych projektów baza danych SQLite jest łatwa w implementacji oraz bardzo dobrze zintegrowana jest z Django.

3.3. HTML, CSS, Bootstrap, TinyMCE

HTML, CSS służą do stworzenia struktury strony internetowej wraz z jej wyglądem. Bootstrap natomiast pozwala na tworzenie responsywnych oraz użytecznych stron, które będą łatwe w obsłudze na różnej wielkości ekranach.

TinyMCE jest rozbudowanym edytorem tekstu umożliwiającym m.in. zmianę stylu tekstu oraz dodawanie tabel, czy zdjęć.

3.4. System kontroli wersji

Przy tworzeniu repozytorium projektu posłużono się oprogramowaniem GIT.

3.5. Google Analytics

Użyto Google Analytics do monitorowania odwiedzin na blogu.

4. Opis modułów.

W rozdziale opisano moduły aplikacji.

Pierwszy moduł aplikacji dotyczy strony głównej, wyszukiwania wpisów oraz szablonu. W nim również są zawarte wszystkie modele. Następnym modułem jest to moduł związany z użytkownikiem, obecnie wykorzystywany jest przy logowaniu użytkownika do systemu. W późniejszym czasie może być wykorzystany do rejestracji oraz zmiany danych użytkowników. Trzeci moduł dotyczy wpisów (artykułów i testów), umożliwia on dodawanie nowych wpisów, ich edycję oraz wyświetlanie. Kolejnym modułem jest strona z kontaktem. Wyświetlone są dane kontaktowe oraz formularz, który umożliwia wysłanie wiadomości. Piąty moduł odpowiedzialny jest za wyświetlenie strony związanej z osobą prowadzącą bloga. Znajduje się tam opis blogera. Ostatnim modułem jest galeria, która wyświetla wszystkie zdjęcia dodane do wpisów.

Każdy z modułów umożliwia edycję danych oraz stylu wyświetlania, właściwości te, można zmienić na utworzonych stronach internetowych bądź też w panelu administracyjnym. Aby zmienić te dane, należy być zalogowanym użytkownikiem.

5. Instrukcja obsługi

W rozdziale opisano obsługę aplikacji.

Użytkownik po zalogowaniu się w serwisie (górny prawy przycisk „Sign in”) ma możliwość na stronie głównej zmienić treść nagłówka. Przechodząc do strony „Articles” lub „Tests” użytkownik może dodać artykuł bądź test oraz przeglądać wpisy. Po wejściu na dany wpis można zmienić kolor tła, zdjęcie główne, czy wpis ma się wyświetlać na głównej stronie, czy ma być widoczny, ustalić datę, autora, tytuł oraz treść wpisu. Na stronie „Contact” jest możliwość zmiany etykiet i wartości związanych z numerem telefonu oraz adresem e-mail. Tutaj również można wysłać wiadomość do administracji. W zakładce „About me” jest możliwość zmiany nagłówka oraz treści tekstu. W galerii jest możliwość przeglądania zdjęć. Strona umożliwia również wyszukiwanie wpisów po tytule wpisów.

6. Instrukcja instalacji

W celu zainstalowania aplikacji, należy uruchomić środowisko wirtualne znajdujące się w folderze „myenv”, a następnie przechodząc do folderu „projekt” wpisać w wierszu poleceń „python manage.py runserver”.

7. Testy aplikacji

W tym rozdziale przedstawiono scenariusze testów wraz z wynikami.

- Logowanie z użyciem nieprawidłowych danych
Podczas testu wprowadzono hasło nieprawidłowe do konta użytkownika. Test zakończono pomyślnie, ponieważ użytkownik nie został zalogowany.
- Zmiana treści stopki po 2 minutach bezczynności

Użytkownik zalogowany nie wykonywał żadnych działań przez 2 minuty, po czym chciał zmienić treść stopki. Test wykonał się pomyślnie, ponieważ nie udało się zmienić treści stopki oraz użytkownik został automatycznie wylogowany.

8. Repozytorium

Repozytorium znajduje się na platformie GIT, link do repozytorium:
<https://github.com/DawidWiktor/projektSZT>.