

WOJCIECH KLUSEK
Numer albumu: 083559

PROJEKT I IMPLEMENTACJA SYSTEMU
OBSŁUGI BIBLIOTEKI W OPARCIU
O TECHNOLOGIE: DJANGO, BOOTSTRAP,
HTML, CSS, JAVASCRIPT, SQL.

Praca dyplomowa inżynierska
na kierunku Informatyka

Opiekun pracy dyplomowej:
dr inż. Jacek Wilk-Jakubowski
Zakład Informatyki

Kielce, 2019

Projekt i implementacja systemu obsługi biblioteki w oparciu o technologie: Django, Bootstrap, HTML, CSS, JavaScript, SQL.

Streszczenie

Tematem pracy jest projekt i implementacja systemu służącego do obsługi biblioteki. Główną funkcjonalnością jest rejestrowanie zdarzeń dotyczących wypożyczania oraz zwrotu książek lub czasopism. Projekt pozwala na rejestrację użytkowników w systemie oraz obsługę zarejestrowanych kont przez bibliotekarza. Możliwe jest również dodawanie, modyfikowanie oraz usuwanie wprowadzonych do systemu danych, dotyczących udostępnianych przez bibliotekę zasobów. System ma za zadanie wspomagać bibliotekarza w jego pracy.

Słowa kluczowe:

- Aplikacja webowa
- Biblioteka
- Python
- Framework Django
- Bootstrap
- JavaScript

Spis treści

1. Charakterystyka głównych założeń pracy.	10
2. Założenia dotyczące funkcjonalności systemu.	11
3. Budowa interfejsu użytkownika.	13
4. Realizacja części <i>Front-End</i> w technologiach: Bootstrap, HTML, CSS, JavaScript.	16
4.1. Opis działania oraz wykorzystania technologii Bootstrap.	16
4.2. Wykorzystanie języka JavaScript.	17
5. Realizacja części <i>Back-End</i> w technologiach: Django, SQL.	19
5.1. Przedstawienie frameworka Django.	19
5.2. Omówienie działania wzorca model-template-view.	20
5.3. Omówienie modelu bazy danych wygenerowanego przy pomocy frameworku Django.	25
6. Opis interfejsu użytkownika.	27
6.1. Wstęp.	27
6.2. Opis funkcjonalności dostępnych dla niezalogowanego użytkownika.	28
6.3. Opis funkcjonalności dostępnych dla zalogowanego użytkownika.	32
6.4. Opis funkcjonalności dostępnych dla bibliotekarza.	36
7. Uwagi i wnioski.	46
7.1. Możliwości dalszej rozbudowy projektu.	46
7.2. Podsumowanie.	47
Bibliografia	48

1 Charakterystyka głównych założeń pracy.

Celem pracy jest projekt i implementacja systemu służącego do obsługi biblioteki, zajmującej się wypożyczaniem książek i czasopism. Praca została zaimplementowana w oparciu o technologie: Django, Bootstrap, HTML, CSS, JavaScript oraz SQL.

Część *Front-End* odpowiada za prezentację danych oraz wygląd interfejsu użytkownika, wykorzystano tutaj framework Bootstrap w wersji czwartej. Bootstrap pozwala tworzyć responsywny interfejs użytkownika. Dopełnieniem wykorzystanej technologii Bootstrap jest język JavaScript, wykorzystany w projekcie do tworzenia animacji na stronie oraz dynamicznie generowanych wykresów.

Część *Back-End* odpowiada za przetwarzanie danych oraz pozwala na implementację funkcjonalności systemu. Wykorzystany framework Django pozwala, za pomocą klas modeli, widoków oraz systemu szablonów, utworzyć aplikację internetową. Dane przetworzone w części *Back-End* mogą zostać umieszczone, przy pomocy odpowiednich znaczników udostępnianych przez framework Django w części *Front-End*.

Głównym założeniem pracy było utworzenie systemu wspomagającego oraz usprawniającego pracę biblioteki, pozwalającego na rejestrowanie wypożyczeń i obsługę użytkowników.

2 Założenia dotyczące funkcjonalności systemu.

Projekt posiada zaimplementowane następujące funkcjonalności:

- Wypożyczanie oraz zwrot zasobów biblioteki przez czytelników.

Zasobami biblioteki są książki oraz czasopisma dostępne fizycznie w bibliotece. Informacje o fizycznych zasobach biblioteki są wprowadzane do systemu przez bibliotekarza. Dzięki temu możliwe jest informowanie użytkownika np. o liczbie dostępnych do wypożyczenia egzemplarzy zasobu. Bibliotekarz ma możliwość odnotowania faktu wypożyczenia lub zwrotu fizycznego egzemplarza książki, lub czasopisma przy wykorzystaniu systemu. System biblioteczny zapisuje informacje o danym wypożyczeniu, są to m.in.: dane bibliotekarza, który wypożyczał egzemplarz czytelnikowi, dane czytelnika, który ten egzemplarz wypożyczył, data wypożyczenia oraz data zwrotu fizycznego egzemplarza książki lub czasopisma do biblioteki.

- System kar nakładanych na użytkowników.

System pozwala wyliczać oraz nakładać kary na osoby wypożyczające. Nałożenie kary następuje w momencie przekroczenia daty zwrotu egzemplarza do biblioteki. Przy pomocy napisanego skryptu możliwe jest określenie na podstawie daty bieżącej oraz daty zwrotu egzemplarza, czy została przekroczona data zwrotu do biblioteki. Bazując na liczbie dni spóźnienia oraz określonej kwocie naliczanej osobno dla każdej książki, obliczana jest kara finansowa przypisywana do konta osoby wypożyczającej. Informacje o nałożonych karach są widoczne zarówno dla osoby której, one dotyczą, jak i dla wszystkich bibliotekarzy.

Projekt w obecnym stanie nie posiada systemu płatności pozwalającego na opłacenie należności bezpośrednio na stronie. Nie ma możliwości przypisania kolejnych wypożyczeń do konta, które posiada wypożyczenia z przekroczonym terminem zwrotu. Zwrot fizycznych egzemplarzy zasobów w bibliotece oraz usunięcie przez bibliotekarza przypisanych do czytelnika wypożyczeń z przekroczonym terminem zwrotu skutkuje przywróceniem możliwości przypisania do jego konta nowych książek lub czasopism.

- Rejestracja i logowanie do systemu. Nadawanie uprawnień bibliotekarza.

Projekt pozwala na rejestrację użytkowników w systemie. Po poprawnym uzupełnieniu formularza i rejestracji konto użytkownika pozostaje nieaktywne, aktywacji konta dokonuje użytkownik o statusie bibliotekarza obecny od początku użytkowania systemu (konto bibliotekarza może być utworzone przy pomocy skryptu frameworka Django). Po aktywacji konta użytkownik może się na nie zalogować. System udostępnia także funkcjonalność resetowania hasła w oparciu o wysyłane wiadomości e-mail. Zawierają one wygenerowany link prowadzący do formularza resetowania hasła.

Uprawnienia bibliotekarza mogą zostać przypisane do istniejącego w systemie konta czytelnika, prawa te mogą być także odebrane. Ważnym udogodnieniem jest to, że konto bibliotekarza oprócz rozszerzenia uprawnień zachowuje też funkcjonalność konta czytelnika. Bibliotekarz może wypożyczać książki innym bibliotekarzom, jak również przeglądać swoje wypożyczenia.

- Dodawanie, modyfikowanie oraz usuwanie danych zgromadzonych w tabelach.

Użytkownik o statusie bibliotekarza ma możliwość, przy wykorzystaniu zaimplementowanych formularzy, operowania na danych zgromadzonych w systemie. Bibliotekarz może dodawać, modyfikować oraz usuwać: książki lub czasopisma, oraz ich egzemplarze, autorów, kategorie oraz wydawnictwa. Każda próba usunięcia danych z systemu poprzedzona jest stroną, na której należy potwierdzić wykonywaną akcję.

- Zabezpieczenie dostępu do poszczególnych części systemu.

Liczba funkcjonalności, które mogą zostać wykorzystane przez osobę korzystającą z systemu, jest zależna od dwóch czynników. Pierwszy z nich dotyczy tego, czy odwiedzający posiada konto w systemie oraz, czy jest zalogowany. Drugi czynnik dotyczy praw, jakie posiada zalogowany użytkownik.

W systemie wyróżniamy konto użytkownika (czytelnika), pozwalające np. na wgląd do listy wypożyczonych przez niego egzemplarzy książek lub czasopism. Konto z podwyższonymi uprawnieniami (konto bibliotekarza) pozwala na pełne zarządzanie danymi dotyczącymi zasobów biblioteki, jak również zarządzanie kontami zarejestrowanych użytkowników. Osoba korzystająca z systemu, ale nie posiadająca w nim konta, może natomiast przejrzeć listę udostępnianych w bibliotece zasobów, wraz z liczbą oraz statusem dostępności egzemplarzy tych zasobów. Może także zapoznać się z informacjami które dotyczą: autorów, wydawnictw oraz kategorii zasobów.

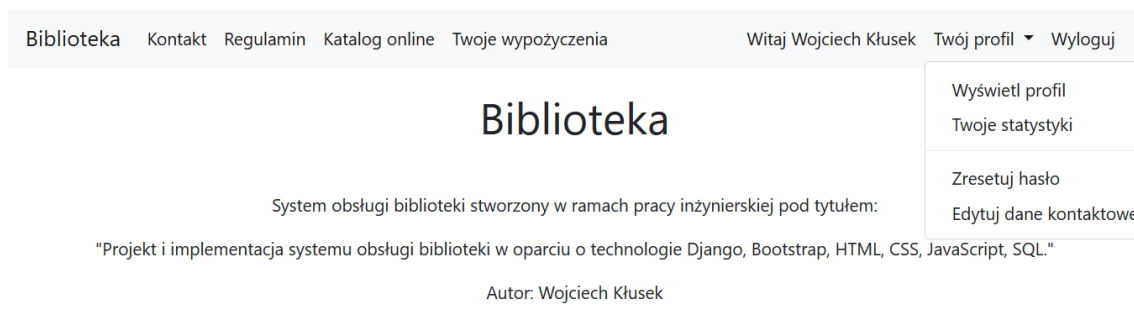
3 Budowa interfejsu użytkownika.

Rozdział ten opisuje w sposób ogólny, działanie zaprojektowanego interfejsu użytkownika. Bardziej szczegółowy opis działania systemu został zaprezentowany w rozdziale 6. Zawartość interfejsu zmienia się w zależności od tego, czy korzysta z niej osoba zalogowana, czy niezalogowana oraz od praw, jakie przypisane są do konta zalogowanego użytkownika. Stałym elementem interfejsu jest menu główne znajdujące się u góry strony.



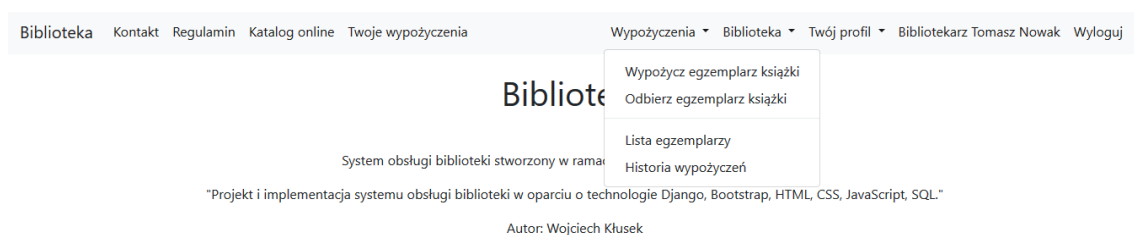
Rysunek 3.1: Fragment strony głównej. Widoczne menu główne wyświetlane dla niezalogowanego użytkownika.

Menu główne wyświetlane dla niezalogowanego użytkownika (*Rys. 3.1*) pozwala na dostęp do Katalogu online, gdzie możliwe jest zapoznanie się z udostępnianymi przez bibliotekę zasobami. Odpowiednie odnośniki pozwalają na dostęp do regulaminu biblioteki oraz informacji kontaktowych. Elementy umieszczone w prawej części menu głównego pozwalają zarejestrować się w systemie lub zalogować do niego.



Rysunek 3.2: Fragment strony głównej. Widoczne menu główne wyświetlane dla zalogowanego użytkownika (czytelnika).

Po zalogowaniu użytkownik otrzymuje dostęp do dodatkowych funkcjonalności dostępnych z poziomu menu głównego (*Rys. 3.2*). Zalogowany użytkownik po kliknięciu w element "Twoje wypożyczenia" ma możliwość wglądu do: listy wypożyczonych przez niego egzemplarzy, tabeli nałożonych kar oraz swojej historii wypożyczeń. Po prawej stronie menu głównego widoczne jest: imię i nazwisko zalogowanego użytkownika, rozwijane podmenu zawierające odnośniki do dodatkowych stron oraz link pozwalający na wylogowanie się.



Rysunek 3.3: Fragment strony głównej. Widoczne menu główne wyświetlane dla zalogowanego użytkownika, o statusie bibliotekarza.







Posiadając konto z uprawnieniami bibliotekarza użytkownik uzyskuje dostęp do wszystkich funkcjonalności systemu. Dodatkowo bibliotekarz zachowuje możliwość przypisywania wypożyczeń do swojego konta. Menu główne (*Rys. 3.3*) oprócz opisanych wcześniej funkcjonalności pozwala na dostęp do dodatkowych stron. Podmenu 'Wypożyczenia' grupuje linki do stron, które odpowiadają za wypożyczanie oraz zwrot egzemplarzy książek lub czasopism. Możliwy jest dostęp do tabeli wszystkich aktywnych w danej chwili wypożyczeń oraz tabeli zawierającej wypożyczenia zwrócone do biblioteki.

Drugie podmenu, "Biblioteka" daje bibliotekarzowi dostęp do tabel opisujących dane dotyczące zasobów biblioteki oraz dostęp do tabeli z informacjami o kontach zarejestrowanych użytkowników. Dla przykładu klikając na element podmenu "Autorzy" wyświetlana jest lista wszystkich autorów, których zasoby udostępnia biblioteka (*Rys. 3.4*). Z poziomu tej podstrony możliwy jest też dostęp do formularza pozwalającego dodać informacje o nowym autorze.

Autorzy



[Dodaj autora](#)

Imię i/lub nazwisko ▾

Opcje	Imię i nazwisko	Narodowość	Liczba książek
 	David Ascher	USA	1
 	Aidas Bendraitis	USA	1
 	Agnieszka Ciborowska	Polska	1




Rysunek 3.4: Tabela opisuje autorów, których książki lub czasopisma są zgromadzone w bibliotece.

Każda z tabel umieszczonych w systemie posiada formularz, za pomocą którego możliwe jest filtrowanie wyświetlanych w niej informacji. Ważnym elementem tabel jest kolumna opcji. Jeśli występuje, pozwala na wykonanie opisanych w legendzie akcji. Dla przykładu klikając na pierwszą ikonę w kolumnie opcji (*Rys. 3.4*) możliwe jest zapoznanie się ze zgromadzonymi w systemie informacjami o danym autorze. Każda tabela posiada także zaimplementowany mechanizm paginacji. Jeśli tabela zawiera w sobie odpowiednią ilość wierszy, wyświetlany wówczas paginator dzieli zawartość wyświetlaną w tabeli na strony (*Rys. 3.5*).

 	Mizanur Rahman	Bangladesz	1
---	----------------	------------	---

« 1 2 »

Legenda:

-  Szczegóły autora
-  Edytuj autora
-  Usuń autora (gdy nie posiada przypisanych książek)

Rysunek 3.5: Podziału zawartości tabeli na strony po których można nawigować.

4 Realizacja części *Front-End* w technologiach: Bootstrap, HTML, CSS, JavaScript.

4.1 Opis działania oraz wykorzystania technologii Bootstrap.

Bootstrap jest to framework technologii CSS, dzięki któremu możliwe jest tworzenie interfejsu graficznego stron internetowych. W projekcie użyto frameworka Bootstrap w wersji czwartej. Technologia ta opiera się na wykorzystaniu HTML, wykorzystaniu arkuszy stylu CSS oraz (opcjonalnie) elementów języka JavaScript. Korzystanie z Bootstrap może odbywać się na dwa sposoby. Pierwszy z nich polega na dodaniu do sekcji `<head></head>` dokumentu HTML, odnośników do biblioteki Bootstrap umieszczonej na stronie twórców. Drugim sposobem jest pobranie plików Bootstrap bezpośrednio na dysk, oraz odwołanie się do nich we wspomnianej wcześniej sekcji `<head></head>` dokumentu HTML.

W projekcie zdecydowano się na drugie rozwiązanie, dzięki temu wszystkie pliki niezbędne do prawidłowej pracy projektu są dostępne lokalnie.

Działanie frameworka Bootstrap polega na podziale strony na siatkę [3] zawierającą 12 kolumn. Definiując element np. pole tekstowe możliwe jest przypisanie mu rozmiaru na podstawie liczby kolumn, jaką element będzie zajmował na stronie. Korzystając z przygotowanych we frameworku kontenerów utworzono poszczególne sekcje strony. Strona może zostać podzielona na wiele sekcji, mogą być one w sobie także zagnieżdżone. Zaimplementowany w technologii Bootstrap interfejs siatkowy sprawia, że kolumny dynamicznie dopasowują się do każdej zmiany rozmiaru okna przeglądarki, co sprawia, że interfejs staje się responsywny [6].

Bootstrap dostarcza gotową implementację elementów strony internetowej, ich użycie polega na umieszczeniu wewnątrz znacznika HTML nazwy odpowiedniej klasy. Możliwe jest np. wyświetlenie na stronie tabeli, poprzez wykorzystanie odpowiedniej nazwy klasy frameworka Bootstrap w atrybucie "class" znacznika `<table></table>` (*Rys. 4.1*). Pozwala to wygenerować tabelę, która posiada ustalony styl oraz wygląd (*Rys. 4.2*).

```

<table class="table table-bordered table-hover">
  <thead class="thead-light">
    <tr>
      <th scope="col">Opcje</th>
      <th scope="col">Imię i nazwisko</th>
      <th scope="col">Narodowość</th>
      <th scope="col">Liczba książek</th>
    </tr>
  </thead>
  ...
</table>

```

Rysunek 4.1: Fragment kodu dokumentu HTML.

W atrybutach "class" znaczników `<table></table>` oraz `<thead></thead>` umieszczono nazwy klas zdefiniowanych we frameworku Bootstrap.

Opcje	Imię i nazwisko	Narodowość	Liczba książek
 	David Ascher	USA	2
 	Aidas Bendoraitis	USA	2

Rysunek 4.2: Wygląd tabeli uzyskany poprzez zastosowanie frameworka Bootstrap.

Przy pomocy technologii Bootstrap zrealizowano elementy statyczne stron, takie jak: menu główne, tło zawartości strony, rozmieszczenie elementów na stronie lub wygląd tabel. Wszystkie pliki HTML są jednocześnie częścią systemu szablonów, czyli rozwiązania udostępnianego przez framework Django. Zasadę działania systemu szablonów, jak również sposób wyświetlania na stronie treści generowanej dynamicznie opisuje rozdział 5.

4.2 Wykorzystanie języka JavaScript.

Język JavaScript został wykorzystany w projekcie poprzez dodanie skryptów niezbędnych do prawidłowego działania animacji menu głównego, utworzonego we frameworku Bootstrap.

Kolejnym elementem projektu, w którym język ten znalazł zastosowanie, są dynamicznie generowane wykresy. Wykresy przedstawiają liczbę dokonanych w ciągu roku wypożyczeń (*Rys. 4.3*).



Rysunek 4.3: Liczba wypożyczeń dokonanych w ciągu danego roku w całej bibliotece, z podziałem na poszczególne miesiące.

Podczas implementacji wykresów wykorzystano dostępną na zasadach open-source bibliotekę Chart.js. Pozwala ona projektować oraz wyświetlać wykresy przy wykorzystaniu języka JavaScript.

```
var ctx = $('#bar-chart-books-per-month');
ctx.height(300);

new Chart(ctx, {
  type: 'bar',
  data: {
    labels: ["Styczeń", "Luty", "Marzec", "Kwiecień", "Maj", "Czerwiec",
            "Lipiec", "Sierpień", "Wrzesień", "Październik", "Listopad", "Grudzień"],
    datasets: [
      {
        label: "Wypożyczeń w miesiącu",
        backgroundColor: ["#6F9FD8", "#6F9FD8", "#BFD641",
                        "#BFD641", "#BFD641", "#FFD662",
                        "#FFD662", "#FFD662", "#DC4C46",
                        "#DC4C46", "#DC4C46", "#6F9FD8"],
        data: {{borrows_per_month_chart_data}}
      }
    ]
  }
});
```

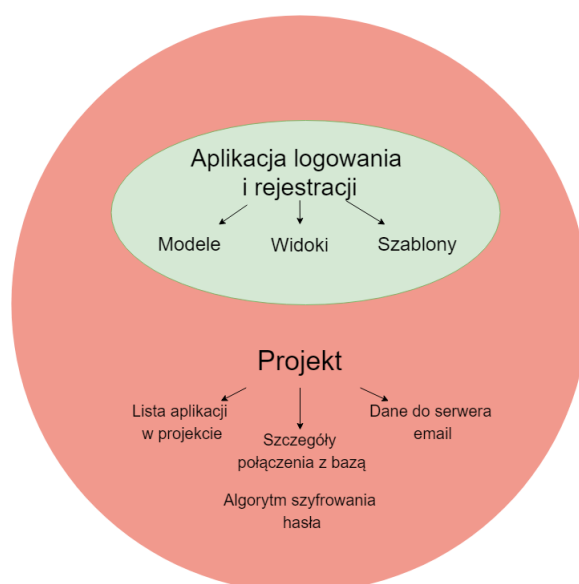
Rysunek 4.4: Fragment kodu odpowiedzialnego za implementację wykresów na stronie.

Rysunek (*Rys. 4.4*) przedstawia fragment kodu, który jest odpowiedzialny za implementację wyświetlanych na stronie wykresów. Atrybut "labels" określa oś x wykresu, są to poszczególne miesiące roku. Możliwe jest określanie koloru słupków wyświetlanych na wykresie za pomocą atrybutu "backgroundColor". Dane umieszczone na wykresie pochodzą z tabeli "Historia wypożyczeń", są one pobierane z bazy danych przez widok zaimplementowany we frameworku Django. Następnie dane te przesyłane są do strony w zmiennej "borrows_per_month_chart_data", przy wykorzystaniu odpowiednich dla frameworku Django znaczników. Dane te widoczne będą na wykresie w momencie przejścia na stronę.

5 Realizacja części *Back-End* w technologiach: Django, SQL.

5.1 Przedstawienie frameworka Django.

Django to framework języka Python przeznaczony do tworzenia aplikacji internetowych. Rozpowszechniany jest on na otwartej licencji. W projekcie użyto wersji 1.11 która zapewnia przedłużony okres wsparcia (wersja LTS). W tej technologii istotny jest podział na projekt oraz aplikację (*Rys. 5.1*). Przy rozpoczęciu pracy z frameworkiem Django należy utworzyć projekt.



Rysunek 5.1: Podział na projekt oraz aplikację w technologii Django.

Projekt definiuje informacje dotyczące np.: rodzaju wykorzystywanej bazy danych, algorytmu szyfrowania hasła, miejsca przechowywania plików HTML lub listy aplikacji. Projekt sam w sobie nie realizuje żadnej funkcjonalności, jest to zadanie aplikacji. Przy projektowaniu aplikacji wykorzystywany jest wzorzec MTV (model-template-view). Projekt może się składać z wielu aplikacji realizujących zaplanowane funkcjonalności. Raz napisana aplikacja może z kolei zostać użyta w kolejnych projektach.

W skład pracy inżynierskiej wchodzi dwie aplikacje: pierwsza z nich realizuje funkcjonalność logowania oraz rejestracji użytkowników, druga odpowiada za funkcjonalności, które dotyczą obsługi biblioteki.

5.2 Omówienie działania wzorca model-template-view.

Wzorzec model-template-view wykorzystywany w technologii Django pozwala na oddzielenie w projekcie warstwy danych, warstwy logiki biznesowej oraz warstwy prezentacji [2]. Działanie wzorca zostanie zaprezentowane na przykładzie z projektu, który dotyczy funkcjonalności wyświetlania listy wydawnictw.

Warstwa modelu odpowiada za definiowanie struktur danych za pomocą klas napisanych w języku Python. Każdy atrybut klasy odpowiada za pole w tabeli, możliwe jest także określenie dodatkowych parametrów np. maksymalna ilość znaków dla pola typu varchar. Django udostępnia w plikach projektu skrypt manage.py [4]. Uruchomienie go z odpowiednim parametrem w konsoli, pozwala zmapować klasy modeli na odpowiadające im tabele w bazie danych (*Rys. 5.2*). Za pomocą klasy modelu można także określić powiązania (np. wiele-do-wielu) między klasami modeli (*Rys. 5.3*). W przypadku relacji wiele-do-wielu wbudowany ORM automatycznie tworzy tabelę łączącą dla obu tabel [1]. Framework Django pozwala na mapowanie klas modeli na tabele dla takich systemów baz danych jak: Oracle, MySQL lub SQLite. Dane dotyczące szczegółów połączenia z bazą określane są w plikach projektu.

```
class PublishingHouse(models.Model):
    name = models.CharField(max_length=400, unique=True)
    city = models.CharField(max_length=300)
    street = models.CharField(max_length=300)
    house_number = models.CharField(max_length=20)
    postal_code = models.CharField(max_length=9)
```

Wydawnictwo	
id	int
name	varchar
city	varchar
street	varchar
house_number	varchar
postal_code	varchar

Rysunek 5.2: Klasa modelu opisuje wydawnictwo. Po prawej stronie widoczny rezultat mapowania klasy modelu na tabele.

```
class Book(models.Model):
    authors = models.ManyToManyField(Author)
```

Rysunek 5.3: Fragment klasy modelu. Tworzenie relacji wiele-do-wielu pomiędzy modelem "Book" i "Author".

Warstwa widoku pozwala operować na danych w tabelach, zdefiniowanych przez modele oraz przekazywać dane pobrane z tabel do szablonów stron. To tutaj znajduje się logika aplikacji. Django udostępnia gotowe rozwiązania, dotyczące implementacji widoków, na potrzeby podstawowych operacji na bazie danych, są to tzw. widoki generyczne. Widok jest funkcją języka Python, która otrzymuje żądanie sieciowe oraz odpowiada na nie, wewnątrz widoku znajduje się kod odpowiadający za zwrot żądanej odpowiedzi. Natomiast widoki generyczne spełniają tę samą funkcjonalność z wykorzystaniem klas. Każda klasa widoku generycznego dziedziczy po klasie "View", która posiada zaimplementowaną obsługę metod HTTP [4].

```
1 class PublishingHouseListView(LoginRequiredMixin, SuperuserRequiredMixin, generic.ListView):
2     login_url = reverse_lazy('no_permission')
3     model = models.PublishingHouse
4     context_object_name = "publishing_houses_list"
5     paginate_by = 10
6     template_name = "library_app/publishinghouse_list.html"
7
8     def get_queryset(self):
9         search_query = self.request.GET.get("search_query")
10        search_type = self.request.GET.get("search_type")
11        query_result = None
12        if search_query is not None:
13            if search_type == "name":
14                query_result = models.PublishingHouse.objects.filter(name__icontains=search_query)
15                return query_result.order_by("name")
16            else:
17                return models.PublishingHouse.objects.all().order_by("name")
```

Rysunek 5.4: Fragment klasy widoku generycznego, który odpowiada za wyświetlanie listy wydawnictw.

Dzięki wykorzystaniu rozwiązania, jakim są widoki generyczne, możliwe jest zaimplementowanie w bardziej efektywny sposób widoków odpowiadających np. za wyświetlanie listy wydawnictw (*Rys. 5.4*). W lini nr 1 widoczne jest dziedziczenie z klasy "View", co jest charakterystyczne dla widoków opartych na klasach generycznych, oprócz tego klasa `PublishingHouseListView` dziedziczy także po domieszkach (*ang. mixins*). Atrybut "model" określa z jakiej klasy modelu, a w konsekwencji, z jakiej tabeli w bazie danych korzystać będzie widok. Metoda `get_queryset(self)` pozwala określić jakie dane mają zostać pobrane z tabeli określonej w atrybucie "model" a następnie przesłane do szablonu strony.

Dla przykładu w lini 17 widoczny jest fragment kodu odpowiadający za zawrót z widoku do szablonu strony wszystkich wydawnictw posortowanych rosnąco po nazwie. Atrybut "context_object_name" określa nazwę, pod jaką dane te będą widoczne w szablonie, natomiast atrybut "template_name" określa plik szablonu, na podstawie którego budowana będzie strona.

Domieszka jest to klasa dziedziczenia wielokrotnego [4]. Klasy *mixin* można wykorzystać do rozszerzenia widoku o dodatkowe funkcjonalności. W tym przypadku dziedziczenie po "LoginRequiredMixin" oraz "SuperuserRequiredMixin" (*Rys. 5.4*) powoduje automatyczne sprawdzanie, czy odwiedzający stronę z listą wydawnictw (za którą odpowiedzialna jest klasa `PublishingHouseListView`) jest zalogowany, oraz czy jego konto ma prawa superużytkownika. Jeśli tak, to strona zostanie mu wyświetlona. Natomiast jeśli odwiedzający nie posiada odpowiednich praw, zostanie przekierowany na stronę określoną w atrybucie "login_url".

Warstwa szablonu (template) odpowiada za prezentację danych przekazanych przez warstwę widoku na stronie. Szablon jest plikiem HTML w którym możliwe jest zawarcie specjalnych znaczników udostępnianych przez framework Django. Dzięki tym znacznikom możliwe jest stosowanie prostych konstrukcji takich jak np. pętle lub instrukcje warunkowe, a w konsekwencji wyświetlanie dynamicznej zawartości na stronie. Dokument HTML, zwracany przez widok, jest generowany z szablonu uzupełnionego o dane pobrane z bazy danych [2].

Przykładem może być wyświetlanie za pomocą pętli listy wydawnictw na stronie (*Rys. 5.5*). W lini nr 1 widoczne jest zastosowanie znacznika pętli `for`. W pętli wyświetlane są wszystkie atrybuty pojedynczego wydawnictwa. Widoczna jest także zmienna "publishing_houses_list", w której znajdują się wszystkie wydawnictwa pobrane z bazy danych przez widok, pokrywa się to z nazwą zmiennej, jaka została określona w atrybucie "context_object_name" widoku generycznego (*Rys. 5.4*).

```
1 {% for publishing_house in publishing_houses_list %}
2     <tr>
3         <td>{{publishing_house.name}}</td>
4         <td>{{publishing_house.street}}</td>
5         <td>{{publishing_house_book_count}}</td>
6     </tr>
7 {% endfor %}
```

Rysunek 5.5: Fragment kodu szablonu.

Framework Django udostępnia także funkcjonalność dziedziczenia szablonów [5]. Możliwe jest utworzenie szablonu bazowego (dokumentu HTML) zawierającego np. tag `<head></head>` wraz z jego zawartością. Tworząc kolejny szablon, należy odwołać się w nim do szablonu bazowego. W ten sposób w momencie przejścia do nowo utworzonej strony załadowane zostaną informacje znajdujące się w szablonie bazowym. Eliminuje to konieczność powtarzania wspólnego kodu w wielu szablonach.

Ostatnim krokiem niezbędnym do prawidłowego wyświetlenia strony jest powiązanie widoku, który odpowiada za warstwę logiki z unikatowym adresem URL. Wzorzec adresu URL składa się z wyrażenia regularnego języka Python, klasy widoku oraz unikatowej nazwy (*Rys. 5.6*). Ostatni parametr, czyli nazwa, jest wykorzystywany w szablonie strony, dzięki czemu możliwe jest utworzenie, w dokumencie HTML, odnośnika kierującego do strony, np. z listą wydawnictw (*Rys. 5.7*).

Po kliknięciu w link framework Django zmapuje nazwę z powiązaniem z nią widokiem, następnie widok ten zostanie zaimportowany przez Django i wykonany [4]. Na podstawie szablonu oraz pobranych przez klasę widoku danych wygenerowana zostanie strona internetowa.

```
url(r'^publishing_houses_list/$',  
    views.PublishingHouseListView.as_view(),  
    name='publishinghouses_list')
```

Rysunek 5.6: Wzorzec adresu URL.

```
<a class="dropdown-item"  
    href="{% url 'library_app:publishinghouses_list' %}">  
    Wydawnictwa  
</a>
```

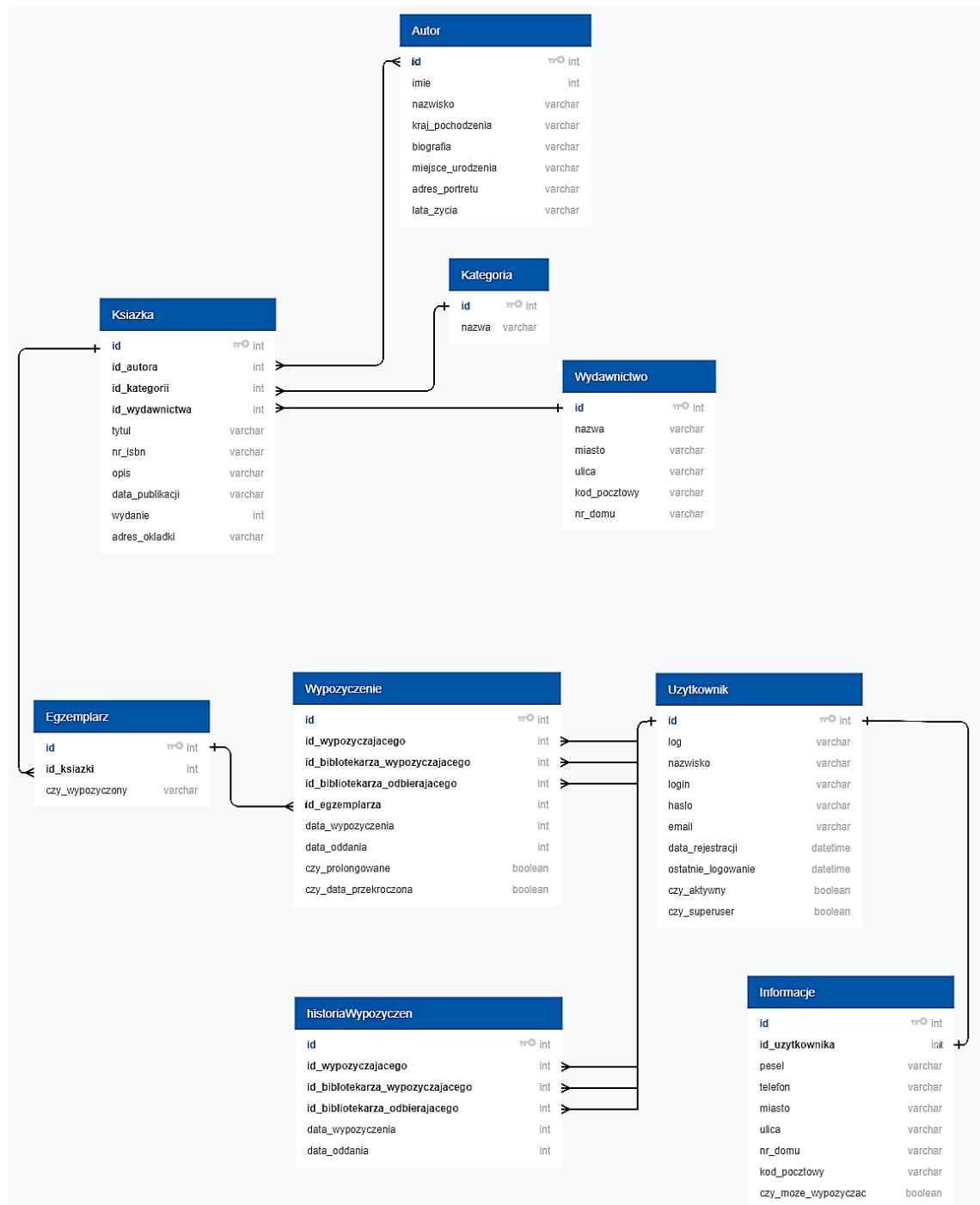
Rysunek 5.7: Odnośnik do strony wykorzystujący zaimplementowany wzorzec adresu URL.

Tak jak zostało to opisane, wzorzec model-template-view pozwala rozdzielić kod aplikacji na:

- warstwę modelu — odpowiadającą za określanie za pomocą klas, oraz mechanizmu ORM tabel w bazie danych,
- warstwę szablonu — definiującą to, w jaki sposób dane zostają wyświetlane na stronie,
- warstwę widoku — opisującą logikę aplikacji. Przy pomocy klas generycznych możliwe jest modyfikowanie oraz pobieranie informacji z bazy danych. Na podstawie szablonu i danych generowana jest strona internetowa.

5.3 Omówienie modelu bazy danych wygenerowanego przy pomocy frameworku Django.

Wykorzystując klasy modeli oraz zaimplementowany we Framework Django mechanizm ORM, utworzono następującą bazę danych (*Rys. 5.8*).



Rysunek 5.8: Schemat wykorzystywanej w projekcie bazy danych.

Główną tabelą w projekcie jest tabela wypożyczeń. Znajdują się w niej informacje o: użytkowniku który wypożycza zasób biblioteki, bibliotekarzu który ten zasób wypożyczył oraz odebrał, jak również informacja o danym zasobie. Dodatkowo w tabeli wypożyczeń znajdują się informacje o dacie wypożyczenia egzemplarza, dacie jego zwrotu, a także to, czy wypożyczenie było prolongowane oraz, czy nie minął termin zwrotu zasobu do biblioteki.

Tabela książka zawiera informacje o kategorii, wydawnictwie oraz autorach, jest ona też kluczem obcym dla tabeli egzemplarz. Pomiedzy tabelą Autor oraz Książka istnieje relacja wiele-do-wielu, dzięki temu możliwe jest przypisanie książki do wielu autorów, jak również każdy autor może być przypisany do wielu książek. Każdy wiersz tabeli Kategoria oraz Wydawnictwo może być kluczem obcym dla wielu książek, natomiast pojedynczy rekord tabeli Książka może posiadać relacje z tylko jednym wydawnictwem i tylko jedną kategorią.

Framework Django udostępnia wbudowaną tabelę użytkowników, posiada ona w sobie pola identyfikujące użytkownika, znajdują się tu także informacje o tym, czy konto użytkownika jest aktywne oraz, czy ma ono status superużytkownika. Na podstawie wartości pól "czy_superuser" oraz "czy_aktywny" z tabeli Użytkownicy, widok generyczny podejmuje decyzje dotyczącą przydzielenia dostępu do danej strony. Wbudowana we framework tabelę Użytkownik nie posiada w sobie wszystkich niezbędnych pól, dlatego też, aby jednoznacznie identyfikować użytkownika, dodano powiązaną z nią tabelę Informacje. Tabela ta zawiera dodatkowe dane dotyczące użytkownika jak np. PESEL, adres oraz numer telefonu.

Rekordy w tabeli Historia Wypożyczeń zawierają informacje analogiczne do tych z tabeli Wypożyczenie. W momencie zwrotu przez czytelnika książki lub czasopisma, bibliotekarz usuwa z systemu informacje o danym wypożyczeniu. Zanim to jednak nastąpi, informacje o tym wypożyczeniu są kopiowane z tabeli Wypożyczenie do tabeli Historia Wypożyczeń. W ten sposób informacje o zwróconych wypożyczeniach są nadal dostępne oraz stanowią źródło danych np. dla wyświetlanych na stronie wykresów.

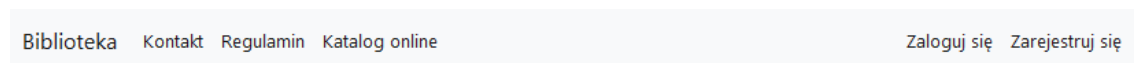
6 Opis interfejsu użytkownika.

6.1 Wstęp.

System ma możliwość określenia, jakiego rodzaju odwiedzający korzysta z systemu. Możliwe są następujące sytuacje:

- Odwiedzający stronę jest niezalogowany,
- Odwiedzający jest zalogowany jako użytkownik nieposiadający praw superużytkownika (czytelnik),
- Odwiedzający jest zalogowany jako użytkownik posiadający prawa superużytkownika (bibliotekarz)

Przy próbie wejścia na daną stronę klasa widoku, która jest odpowiedzialna za przesłanie informacji do szablonu tej strony, sprawdza, która z wymienionych wyżej sytuacji występuje. Dzięki temu możliwe jest blokowanie dostępu do danej strony (*Rys. 6.1*) lub wyświetlanie jej zawartości w przypadku, gdy użytkownik posiada odpowiednie uprawnienia.



Strona dostępna tylko dla administratora.



Rysunek 6.1: Blokowanie dostępu do strony w przypadku braku odpowiednich uprawnień do jej odwiedzenia.

6.2 Opis funkcjonalności dostępnych dla niezalogowanego użytkownika.

Użytkownik niezalogowany ma możliwość dostępu do stron pozwalających na:

- Wyświetlanie katalogu udostępnianych zasobów,
- Rejestrację konta w systemie,
- Logowanie do systemu oraz przypominanie hasła,
- Zapoznanie się z regulaminem oraz informacjami kontaktowymi.

Wyświetlania katalogu udostępnianych zasobów.

Katalog Online						
<input type="text" value="Wyszukaj"/>		<input type="text" value="Tytuł"/>	<input type="button" value="Szukaj"/>			
Opcje	Tytuł	Autor	Kategoria	Wydawnictwo	Egzemplarzy	Wolnych egzemplarzy
	Aplikacje internetowe z Django. Najlepsze receptury	Bendoraitis Aidas	Informatyka	Helion	2	2
	Czysta architektura. Struktura i design oprogramowania. Przewodnik dla profesjonalistów	Martin Robert C.	Informatyka	Helion	1	1

Rysunek 6.2: Strona główna katalogu zasobów udostępnianych przez bibliotekę.

Strona katalogu zasobów wyświetla tabelę (*Rys. 6.2*) zawierającą informacje o wszystkich książkach oraz czasopismach znajdujących się w bibliotece. Wyświetlane są tu takie informacje jak: tytuł książki, autorzy, kategoria i wydawnictwo. Tabela przedstawia także liczbę egzemplarzy danej książki oraz liczbę egzemplarzy możliwych do wypożyczenia. Na stronie znajduje się także formularz, za pomocą którego możliwe jest filtrowanie danych, wyświetlanych w tabeli.

Aby ułatwić nawigowanie po stronie poszczególne elementy w wierszach tabeli zostały zaimplementowane jako linki. Dla przykładu po kliknięciu w dowolną nazwę wydawnictwa, wyświetloną na stronie, użytkownikowi wyświetlana jest kolejna strona, w której zawarte są informacje o tym wydawnictwie, jak również wszystkie zasoby tego wydawnictwa dostępne w bibliotece (*Rys. 6.3*). Pozwala to osobie korzystającej z systemu na wygodne odnajdywanie potrzebnych informacji dotyczących np. książki, którą chciałby wypożyczyć.

Szczegóły wydawnictwa

Nazwa: Helion
Miasto: Warszawa
Adres: Nowa 999
Kod pocztowy: 99-999
Liczba książek wydawnictwa: 9

Powrót

Książki z tego wydawnictwa

Tytuł	Autor	Kategoria	Wydawnictwo	Egzemplarze	Wolnych egzemplarzy
Python. Wprowadzenie	Lutz Mark Ascher David	Informatyka	Helion	1	1
Sieci komputerowe. Kompendium	Krysiak Karol	Informatyka	Helion	1	1
PHP 7. Algorytmy i struktury danych	Rahman Mizanur	Informatyka	Helion	1	0
WordPress dla początkujących	Ciborowska Agnieszka Lipiński Jarosław	Informatyka	Helion	1	1

Rysunek 6.3: Strona zawiera informacje o przykładowym wydawnictwie.

Rejestracja konta w systemie.

Jeśli chodzi o możliwość rejestracji aplikacja udostępnia formularz, za pomocą którego odwiedzający może utworzyć konto w systemie.

Formularz (*Rys. 6.4*) pozwala na wprowadzenie informacji niezbędnych do utworzenia nowego konta, są to między innymi:

- Nazwa użytkownika (login) oraz hasło,
- Adres e-mail, który pozwala na resetowanie hasła,
- Dane osobowe: imię, nazwisko, adres, numer PESEL.

Pola formularza posiadają zaimplementowaną walidację wpisanych danych np. nie istnieje możliwość zarejestrowania się po wpisaniu loginu, który jest już zajęty. Również wpisane hasło jest sprawdzane pod kątem bezpieczeństwa, musi ono zawierać co najmniej 8 znaków oraz być różne od wpisanych w pozostałych polach danych. Po poprawnej rejestracji konto pozostaje nieaktywne, do czasu jego aktywowania przez bibliotekarza. Dopiero wtedy możliwe jest zalogowanie się na nowo utworzone konto.

Zarejestruj się

Użytkownik o tej nazwie już istnieje. ×

Użytkownik

superuser

Użytkownik o tej nazwie już istnieje.
Wymagana. 150 lub mniej znaków. Jedynie litery, cyfry i @/./+/-/_

Adres email

email@email.pl

Imię

Jan

Nazwisko

Kowalski

Hasło

Hasło

Rysunek 6.4: Fragment formularza rejestracji użytkownika. Wyświetlony alert przy niepoprawnej nazwie użytkownika.

Logowanie oraz przypominanie hasła.

Zarejestrowany użytkownik może zalogować się do systemu przechodząc na odpowiednią stronę (*Rys. 6.5*) i wpisując swój login, oraz hasło. Znajduje się tu także przycisk kierujący użytkownika na stronę, za pomocą której może zresetować swoje hasło.

Zaloguj się

Użytkownik

Użytkownik

Hasło

Hasło

Zaloguj się

Zresetuj hasło

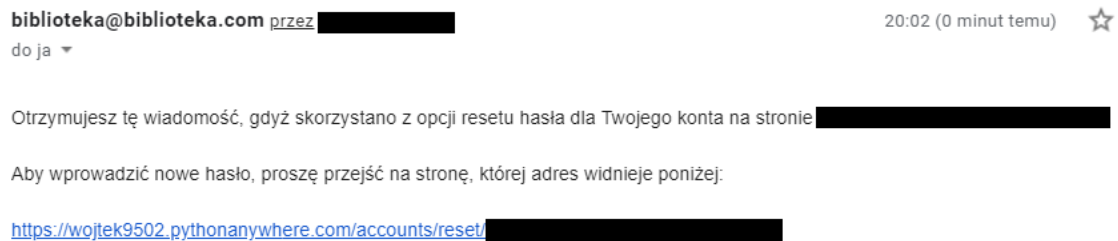
Rysunek 6.5: Formularz logowania.

Użytkownik może skorzystać z funkcjonalności resetowania hasła. Wyświetlony zostaje formularz (*Rys. 6.6*) w którym należy wpisać adres e-mail podany podczas rejestracji. Jeśli w systemie znajduje się użytkownik przypisany do tego adresu, wysyłana jest mu wiadomość (*Rys. 6.7*) zawierający link. Po kliknięciu w link użytkownik może zresetować swoje hasło (*Rys. 6.8*). W projekcie wykorzystano zewnętrzny serwer pocztowy obsługujący wysyłanie wiadomości email.



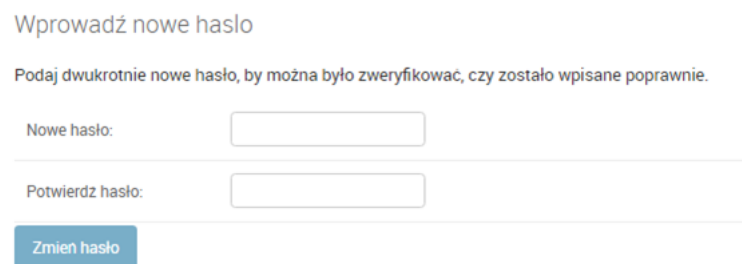
Formularz resetowania hasła. Na górze znajduje się tekst: "Nie pamiętasz swojego hasła? Wprowadź w poniższym polu". Poniżej jest etykieta "Adres email:" i pole tekstowe. Na dole znajduje się przycisk "Zresetuj moje hasło".

Rysunek 6.6: Formularz wpisywania adresu e-mail.



Wiadomość email z tematem "biblioteka@biblioteka.com przez [redacted] do ja ▾". Data i godzina: "20:02 (0 minut temu)" z ikoną gwiazdy. Treść wiadomości: "Otrzymujesz tę wiadomość, gdyż skorzystano z opcji resetu hasła dla Twojego konta na stronie [redacted]". Następnie: "Aby wprowadzić nowe hasło, proszę przejść na stronę, której adres widnieje poniżej:". Na końcu link: "[https://wojtek9502.pythonanywhere.com/accounts/reset/](\"https://wojtek9502.pythonanywhere.com/accounts/reset/\") [redacted]"

Rysunek 6.7: Treść wiadomości wysłanej przez system w celu zresetowania hasła.



Formularz zmiany hasła. Tytuł: "Wprowadź nowe hasło". Instrukcja: "Podaj dwukrotnie nowe hasło, by można było zweryfikować, czy zostało wpisane poprawnie." Dwa pola tekstowe: "Nowe hasło:" i "Potwierdź hasło:". Na dole przycisk "Zmien hasło".

Rysunek 6.8: Formularz resetowania hasła.

Regulamin oraz informacje kontaktowe.

Każdy odwiedzający stronę ma możliwość zapoznania się z treścią regulaminu biblioteki, oraz informacjami kontaktowymi.

6.3 Opis funkcjonalności dostępnych dla zalogowanego użytkownika.

Użytkownik zalogowany oprócz dostępu do wcześniej opisanych funkcjonalności ma dodatkowo możliwość:

- Wyświetlania danych dotyczących konta oraz edytowania swoich danych kontaktowych,
- Wyświetlania listy aktualnie wypożyczonych zasobów oraz prolongowania ich,
- Wyświetlania historii wypożyczeń,
- Zapoznania się z nałożonymi na użytkownika karami,
- Wyświetlania wykresu wypożyczeń dokonanych w danym roku.

Wyświetlanie oraz edytowanie danych użytkownika.

Po wybraniu odpowiedniej opcji w menu przez użytkownika, wyświetlana jest strona zawierająca formularz (*Rys. 6.9*), pozwalający zmienić dane kontaktowe wprowadzone do systemu podczas procesu rejestracji. Użytkownik nie może zmienić swojego adresu e-mail, ponieważ jest z nim związana funkcjonalność przypominania hasła. Adres e-mail przypisany do konta może zmienić jedynie bibliotekarz.

The screenshot shows a web interface for editing contact data. At the top, there is a navigation bar with links: Biblioteka, Kontakt, Regulamin, Katalog online, and Twoje wypożyczenia. On the right side of the bar, it says 'Witaj Wojciech Klusek' followed by a dropdown arrow, 'Twój profil', and 'Wyloguj'. The main heading is 'Edytuj dane kontaktowe'. Below this, there are five input fields, each with a label to its left: 'Ulica' (containing 'Nowa'), 'Numer domu' (containing '999'), 'Kod pocztowy' (containing '99-999'), 'Miasto' (containing 'Kielce'), and 'Telefon' (containing '999888777'). At the bottom left of the form is a blue button labeled 'Edytuj'.

Rysunek 6.9: Formularz edycji danych kontaktowych zalogowanego użytkownika.

Użytkownik może także wyświetlać informacje o swoim profilu (*Rys. 6.10*), wraz z liczbą wypożyczonych książek, datą ostatniego logowania oraz datą rejestracji w systemie.

Twój profil

Imię i nazwisko: Wojciech Kłusek
Email: wojtek@gmail.com
Telefon: 999888777
PESEL: 9999999997
Adres: Nowa 999 99-999 Kielce

Liczba wypożyczeń: 1
Dołączyłeś/aś dnia: 21 listopada 2018 17:43
Ostatnie logowanie dnia: 23 grudnia 2018 18:44

Rysunek 6.10: Dane opisują profil przykładowego użytkownika.


Wyświetlanie listy wypożyczeń oraz funkcjonalność prolongowania.

Poprzez wybór odpowiedniej opcji w menu strony użytkownik uzyskuje dostęp do tabeli (*Rys. 6.11*) zawierającej aktualnie wypożyczone przez niego książki lub czasopisma. Tabela wyświetla wszystkie informacje o wypożyczeniu takie jak: tytuł oraz autorzy wypożyczonego zasobu, dane bibliotekarza wypożyczającego, datę wypożyczenia i oddania zasobu do biblioteki oraz informacje określającą czy dany egzemplarz jest prolongowany. Dodatkowo wiersz wypożyczenia, dla którego minęła data oddania do biblioteki jest oznaczony odmiennym kolorem.

Twoje wypożyczenia

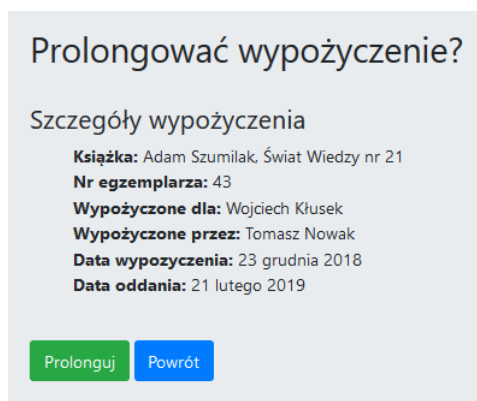
Opcje	Książka	Nr egzemplarza	Wypożyczone przez	Data wypożyczenia	Data oddania	Czy prolongowany?
	Mizanur Rahman PHP 7. Algorytmy i struktury danych	51	Tomasz Nowak	29 listopada 2018	28 listopada 2018	Nie
	Adam Szumilak Świat Wiedzy nr 21	43	Tomasz Nowak	23 grudnia 2018	21 lutego 2019	Nie

Legenda:

 Prolonguje egzemplarz (30 dni)

Rysunek 6.11: Tabela zawiera informację o aktualnie wypożyczonych przez użytkownika zasobach biblioteki.

Każdy egzemplarz zasobu (książki lub czasopisma) wypożyczony z biblioteki może być jednokrotnie prolongowany, jednak tylko wtedy, gdy nie została przekroczona data oddania go do biblioteki. Prolongowanie odbywa się poprzez kliknięcie w odpowiednią ikonę w kolumnie tabeli. Wyświetla się wówczas strona (*Rys. 6.12*) na której należy potwierdzić wykonywaną akcję. Prolongowanie wydłuża datę oddania zasobu do biblioteki o 30 dni.



Prolongować wypożyczenie?

Szczegóły wypożyczenia

Książka: Adam Szumilak, Świat Wiedzy nr 21
Nr egzemplarza: 43
Wypożyczone dla: Wojciech Klusek
Wypożyczone przez: Tomasz Nowak
Data wypożyczenia: 23 grudnia 2018
Data oddania: 21 lutego 2019

[Prolonguj](#) [Powrót](#)

Rysunek 6.12: Przykład strony, która pozwala potwierdzić zamiar prolongowania danego egzemplarza.

Wyświetlanie historii wypożyczeń.

Tabela historii wypożyczeń (*Rys. 6.13*) zawiera dane na temat wypożyczonych w przeszłości przez zalogowanego użytkownika zasobów biblioteki. Znajduje się tu informacja o bibliotekarzu, który odebrał egzemplarz książki lub czasopisma od czytelnika oraz data oddania wypożyczenia.

Historia wypożyczeń					
Książka	Nr egzemplarza	Wypożyczone przez	Odebrane przez	Data wypożyczenia	Data oddania
Adam Mickiewicz Dziady cz2	44	Tomasz Nowak	Tomasz Nowak	21 listopada 2018	21 listopada 2018
Agnieszka Ciborowska Jarosław Lipiński WordPress dla początkujących	52	Tomasz Nowak	Tomasz Nowak	28 listopada 2018	30 listopada 2018
Adam Szumilak Świat Wiedzy nr 21	43	Tomasz Nowak	Tomasz Nowak	21 listopada 2018	2 grudnia 2018

Rysunek 6.13: Tabela z informacjami o historii wypożyczeń zalogowanego użytkownika.

Kary nakładane na użytkownika.

W przypadku gdy czytelnik nie zwróci wypożyczenia na czas, nakładane są na niego kary finansowe. Tabela należności (*Rys. 6.14*) informuje użytkownika o wysokości nałożonej na niego kary finansowej. Kara ta jest zależna od liczby dni spóźnienia, jakie upłynęły od daty zwrotu wypożyczenia. Dla każdego takiego wypożyczenia kara naliczana jest niezależnie.

Projekt w obecnym stanie nie posiada zaimplementowanej funkcjonalności opłacania kar finansowych online, dlatego dane w tabeli należności mają jedynie charakter informacyjny, a sama należność musiałaby być w tej sytuacji opłacana przez czytelnika poza systemem.

Każdy użytkownik, który posiada wypożyczenie z przekroczonym terminem oddania do biblioteki, ma blokową możliwość dalszego wypożyczania. Bibliotekarz nie ma możliwości wypożyczenia tej osobie następnych zasobów biblioteki do momentu uregulowania należności oraz zwrotu egzemplarzy książek lub czasopism posiadających przekroczoną datę zwrotu do biblioteki.

Należności

Egzemplarz	Data oddania	Dni spóźnienia	Kwota za dzień	Kwota
PHP 7. Algorytmy i struktury danych	28 listopada 2018	25	0,5 zł	12,5 zł
			Suma:	12,5 zł

Rysunek 6.14: Tabela zawiera informacje o należnościach naliczonych za książki, które nie zostały zwrócone do biblioteki w terminie.

Wyświetlanie wykresu wypożyczeń.

Każdy użytkownik ma możliwość zapoznania się z tworzonym dla niego dynamicznie wykresem (*Rys. 6.15*). Wykres ten przedstawia wszystkie wypożyczenia zwrócone przez użytkownika w danym roku z podziałem na poszczególne miesiące. Podobny wykres znajduje się na stronie głównej biblioteki, dotyczy on w tym przypadku wszystkich wypożyczeń zwróconych przez wszystkich użytkowników systemu.



Rysunek 6.15: Wykres przedstawia wypożyczenia zwrócone do biblioteki przez przykładowego użytkownika w okresie roku, z podziałem na poszczególne miesiące.

6.4 Opis funkcjonalności dostępnych dla bibliotekarza.

Zalogowany użytkownik posiadający status bibliotekarza ma możliwość dostępu do wszystkich wcześniej opisanych funkcjonalności strony, oznacza to także, że użytkownik o statusie bibliotekarza ma możliwość przypisania do swojego konta wypożyczeń. Dodatkowo konto bibliotekarza pozwala na:

- Rejestrowanie w systemie faktu wypożyczenia i zwrotu zasobu biblioteki,
- Dodawanie, usuwanie oraz modyfikacje danych zawartych w systemie,
- Zarządzanie kontami zarejestrowanych użytkowników,
- Dostęp do panelu administracyjnego.

Rejestrowanie wypożyczeń oraz zwrotów zasobów biblioteki.

Użytkownik posiadający prawa bibliotekarza ma możliwość rejestrowania faktu wypożyczenia książki lub czasopisma użytkownikowi systemu. Po przejściu na odpowiednią stronę z poziomu menu, wyświetlana jest tabela (*Rys. 6.16*). Tabela zawiera wszystkie egzemplarze książek lub czasopism, które można wypożyczyć. Znajduje się tu również formularz, za pomocą którego możliwe jest filtrowanie wyświetlanych danych. Kliknięcie przycisku "Wypożycz" w wybranym wierszu system przenosi bibliotekarza do następnej strony. Znajduje się tam formularz (*Rys. 6.17*) w którym należy wybrać osobę, której egzemplarz książki lub czasopisma będzie wypożyczony oraz kliknąć odpowiedni przycisk.

Wypożycz egzemplarz książki

Wyszukaj egzemplarz książki dla użytkownika

Powrót

Autor	Książka	ISBN	Nr egzemplarza	Akcje
Aidas Bendoraitis	Aplikacje internetowe z Django. Najlepsze receptury	3-598-21527-4	55	<input type="button" value="Wypożycz"/>
Aidas Bendoraitis	Aplikacje internetowe z Django. Najlepsze receptury	3-598-21527-4	59	<input type="button" value="Wypożycz"/>

Rysunek 6.16: Fragment tabeli, która umożliwia wypożyczanie zasobu danemu użytkownikowi.

Wypożycz dla

Wojciech Klusek, PESEL: 99999999997

Egzemplarz książki

Aplikacje internetowe z Django. Najlepsze receptury, ISBN: 3-598-21527-4, nr egzemplarza: 55

Data oddania

21.02.2019

Rysunek 6.17: Formularz pozwala wybrać użytkownika, któremu wypożyczony będzie egzemplarz.

W sytuacji, w której czytelnik zwraca fizyczną kopię zasobu do biblioteki, możliwe jest wprowadzenie informacji o tym zdarzeniu do systemu. Na odpowiedniej stronie bibliotekarz może zapoznać się tabelą (*Rys. 6.18*) opisującą wszystkie aktualnie wypożyczane zasoby biblioteczne. Wiersze w tabeli zostały posortowane rosnąco względem daty zwrotu egzemplarza. Nad tabelą znajduje się formularz pozwalający filtrować wyświetlane dane. Wypożyczenia posiadające przekroczoną datę zwrotu, zostały oznaczone oddzielnym kolorem.

Opcje	Wypożyczone dla	Książka	Nr egzemplarza	Wypożyczone przez	Data wypożyczenia	Data oddania
	Wojciech Kłusek	Mizanur Rahman PHP 7. Algorytmy i struktury danych	51	Tomasz Nowak	29 listopada 2018	28 listopada 2018
	Wojciech Kłusek	Adam Szumilak Świat Wiedzy nr 21	43	Tomasz Nowak	23 grudnia 2018	21 lutego 2019
	Wojciech Kłusek	Aidas Bendraitis Aplikacje internetowe z Django. Najlepsze receptury	55	Tomasz Nowak	25 grudnia 2018	21 lutego 2019

Legenda:

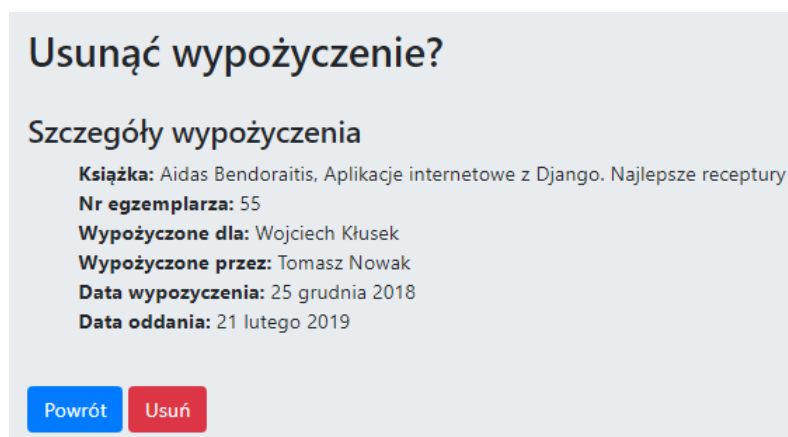
- Przejdź do wypożyczeń użytkownika
- Przejdź do profilu użytkownika
- Usuń egzemplarz z wypożyczonych

Rysunek 6.18: Tabela zawiera informację o wszystkich aktualnie wypożyczonych zasobach bibliotecznych.

Każdy wiersz tabeli (*Rys. 6.18*) posiada powiązane z nim opcje. Bibliotekarz ma możliwość przejścia do strony wyświetlającej wszystkie wypożyczenia danego użytkownika, tam też wyświetlane są wszystkie kary finansowe nałożone na tego czytelnika. Kolejna opcja pozwala przejść do profilu czytelnika, gdzie wyświetlane są jego dane osobowe. Ostatnia opcja pozwala usunąć informacje o danym wypożyczeniu, w takim przypadku egzemplarz nie jest już powiązany z czytelnikiem, który go wypożyczał, jest także dostępny do wypożyczenia dla następnych czytelników. Zwrot zasobu powoduje też przeniesienie informacji o wypożyczeniu do historii wypożyczeń danego użytkownika.

W przypadku gdy czytelnik posiadał wypożyczenie, którego termin zwrotu do biblioteki został przekroczony, blokowana jest mu możliwość dalszego wypożyczania, do momentu, gdy bibliotekarz usunie z systemu wszystkie tego typu wypożyczenia dotyczące danego czytelnika.

Każde usunięcie wypożyczenia w systemie wymaga potwierdzenia wykonywanej akcji na wyświetlanej stronie (*Rys. 6.19*).



Usunąć wypożyczenie?

Szczegóły wypożyczenia

Książka: Aidas Bendoraitis, Aplikacje internetowe z Django. Najlepsze receptury
Nr egzemplarza: 55
Wypożyczone dla: Wojciech Kłusek
Wypożyczone przez: Tomasz Nowak
Data wypożyczenia: 25 grudnia 2018
Data oddania: 21 lutego 2019

Powrót **Usuń**

Rysunek 6.19: Strona pozwala potwierdzić usunięcie wypożyczenia z systemu.

Dodawanie, usuwanie oraz modyfikacje danych zawartych w systemie.

System operuje na danych określonych w tabelach. Jednymi z tabel są: Autor, Książka, Egzemplarz, Wydawnictwo oraz Kategoria. Bibliotekarz może operować na danych zgromadzonych w tych tabelach przy pomocy przygotowanych na stronach formularzy. Dostęp do formularzy możliwy jest przez wybranie odpowiedniej opcji w menu np. chcąc operować na danych z tabeli Autor należy wybrać z menu element "Biblioteka", a następnie "Autorzy".

Formularz dodania książki lub czasopisma (*Rys. 6.20*) pozwala dodać nowy zasób do systemu bibliotecznego. Znajdują się w nim wszystkie niezbędne pola opisujące zasób biblioteczny. Pole określające numer ISBN jest polem unikatowym w tabeli, formularz posiada mechanizm walidacji, który sprawdza unikalność numeru ISBN danej książki lub czasopisma w systemie. Po dodaniu zasobu automatycznie tworzony jest w systemie jego egzemplarz. Z uwagi na relację wiele-do-wielu między tabelami Książka oraz Autor możliwe jest przypisanie do nowo utworzonej książki lub czasopisma wielu autorów.

Autorzy

David Ascher
Aidas Bendoraitis
Agnieszka Ciborowska
Joel Grus

Kategoria

Informatyka

Wydawnictwo

Helion

Numer ISBN

2-12-345678-2

Tytuł

Testowa

Rysunek 6.20: Formularz pozwala na dodanie zasobu do systemu bibliotecznego.

Modyfikowanie istniejącej książki lub czasopisma jest możliwe po wybraniu odpowiedniej opcji w tabeli Katalogu Online (*Rys. 6.21*).

Opcje	Tytuł	Autor	Kategoria	Wydawnictwo	Egzemplarze	Wolnych egzemplarzy
	Aplikacje internetowe z Django. Najlepsze receptury	Bendoraitis Aidas	Informatyka	Helion	2	1

Rysunek 6.21: Fragment tabeli na stronie Katalog Online.

Po wybraniu opcji w wierszu tabeli wyświetlany jest formularz (*Rys. 6.22*) zawierający informacje o zasobie, które można modyfikować, a następnie zapisać dokonane zmiany w systemie.

Tytuł

Aplikacje internetowe z Django. Najlepsze receptury

Autorzy

Ascher David
Ramalho Luciano
Krysiak Karol
Martin Robert C.
Rahman Mizanur

Opis

Nowy opis

Rysunek 6.22: Formularz modyfikacji zasobu bibliotecznego.

Bibliotekarz ma możliwość usunięcia książki lub czasopisma, jednak tylko wtedy gdy nie istnieje w systemie żaden powiązany z usuwanym zasobem egzemplarz. Znajdując się na stronie Katalogu Online bibliotekarz, wykorzystując odpowiednią opcję w wierszu tabeli (*Rys. 6.23*) usuwa dany zasób z systemu potwierdzając swój zamiar kliknięciem odpowiedniego przycisku (*Rys. 6.24*).

Opcje	Tytuł	Autor	Kategoria	Wydawnictwo	Egzemplarzy	Wolnych egzemplarzy
 	Python. Wprowadzenie	Lutz Mark Ascher David	Informatyka	Helion	0	0

Rysunek 6.23: Fragment tabeli na stronie Katalog Online.

Usunąć książkę?

Szczegóły książki

Autor: Mark Lutz David Ascher

Tytuł: Python. Wprowadzenie

Wydawnictwo: Helion

ISBN: 3-598-21519-3

Wydanie: 4

Data publikacji: 1 stycznia 2018

Powrót







Usuń

Rysunek 6.24: Każdy przypadek usunięcia zasobu z systemu wymaga potwierdzenia wykonywanej akcji.

Dodanie egzemplarza książki lub czasopisma polega na wybraniu opcji "Dodaj egzemplarz" widocznej po kliknięciu na przycisk "Biblioteka" w menu głównym. Następnie należy wybrać, dla której książki lub czasopisma ma być utworzony egzemplarz oraz kliknąć przycisk "Dodaj".

Usuwanie egzemplarza jest możliwe tylko wtedy, gdy nie jest on wypożyczony. Po kliknięciu "Wypożyczenia" w menu, należy wybrać opcje "Lista egzemplarzy". Na wyświetlonej stronie bibliotekarz ma możliwość przejrzania listy egzemplarzy wpisanych do systemu (*Rys. 6.25*), znajduje się tu także informacja o tym, czy egzemplarz został wypożyczony, dla kogo został wypożyczony, przez kogo został wypożyczony oraz daty: wypożyczenia oraz zwrotu egzemplarza.

Korzystając z kolumny opcji w tabeli (*Rys. 6.25*), możliwe jest wyświetlenie dodatkowych informacji o egzemplarzu lub usunięcie go z systemu bibliotecznego. Analogicznie do sytuacji, w której usunięciu podlega zasób biblioteczny, usunięcie egzemplarza tego zasobu wymaga potwierdzenia wykonywanej akcji.

Opcje	Autor	Książka	Nr egzemplarza	Pożyczone dla	Pożyczone przez	Data wypożyczenia	Data oddania
 	Adam Mickiewicz	Pan Tadeusz	41	-	-	-	-
 	Juliusz Słowacki	Kordian	42	-	-	-	-
 	Adam Szumilak	Świat Wiedzy nr 21	43	Wojciech Kiusek	Tomasz Nowak	21 listopada 2018	20 stycznia 2019

Rysunek 6.25: Tabela zawierająca wszystkie egzemplarze książek oraz czasopism dostępne w systemie.

W podobny sposób bibliotekarz może usuwać, modyfikować oraz dodawać nowe rekordy do tabel: Wydawnictwo, Kategoria oraz Autor. Operowanie na danych zgromadzonych w tych tabelach jest możliwe dzięki udostępnianym w systemie formularzom. Dostęp do informacji zgromadzonych w wymienionych tabelach odbywa się poprzez kliknięcie elementu "Biblioteka" w menu głównym i wybraniu odpowiedniego linku.

Zarządzanie kontami zarejestrowanych użytkowników.

Użytkownik o statusie bibliotekarza ma możliwość zarządzania kontami użytkowników. Po kliknięciu elementu "Biblioteka" oraz wybraniu linku "Zarządzaj użytkownikami" wyświetlana jest strona zawierająca tabele. Tabela (*Rys. 6.26*) opisuje wszystkich zarejestrowanych w systemie użytkowników. Również z poziomu tej strony możliwe jest przejście do wypożyczeń przypisanych do konkretnego użytkownika.

Zarządzaj użytkownikami

Wyszukaj

Imię i/lub nazwisko

Szukaj

Opcje	Imię i nazwisko	PESEL	Email	Telefon	Liczba wypożyczeń	Czy aktywny	Typ konta
<div><div></div></div>	Tomasz Nowak	9999999996	email@email.com	999999996	0	Tak	Bibliotekarz
<div><div></div><div></div></div>	Wojciech Klusek	9999999997	wojtek@gmail.com	999888777	3	Tak	Czytelnik
<div><div></div><div></div><div></div></div>	Adam Nowak	9999999998	adam@adam.pl	999999998	0	Tak	Czytelnik
<div><div></div><div></div><div></div></div>	Marek Kowalski	9999999999	marek@marek.pl	999999999	0	Nie	Czytelnik

Rysunek 6.26: Fragment tabeli zawiera informacje o zarejestrowanych w systemie użytkownikach.

Wszystkie akcje wykonywane na kontach użytkowników wymagają potwierdzenia na wyświetlanej stronie (*Rys. 6.27*).

Aktywować użytkownika Marek Kowalski?

Imię i nazwisko: Marek Kowalski
Email: marek@marek.pl
Telefon: 999999999
PESEL: 9999999999
Adres: Niwa, 99, 99-999 Kielce

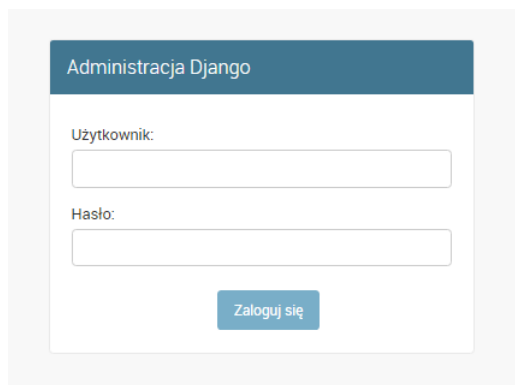
Rysunek 6.27: Potwierdzenie aktywacji konta.

Konto nowego użytkownika może być aktywowane po kliknięciu w odpowiednią opcję. Użytkownik może się wtedy zalogować na swoje konto. Konto użytkownika może być dezaktywowane, jednak tylko wtedy, gdy nie są do niego przypisane żadne wypożyczenia. Dane dotyczące konta pozostają w systemie, jednak nie istnieje wówczas możliwość zalogowania się. Ostatnią opcją jest przekształcenie dotychczasowego konta w konto z prawami bibliotekarza. Posiada ono wtedy pełen dostęp do funkcjonalności udostępnianych przez system.

Aby nie doprowadzić do sytuacji, w której w systemie nie znajduje się żadne konto uprawnione do wypożyczania zasobów bibliotekarz, który jest aktualnie zalogowany do systemu, nie ma możliwości odebrania sobie praw bibliotekarza, może to natomiast zrobić dla pozostałych kont bibliotekarzy.

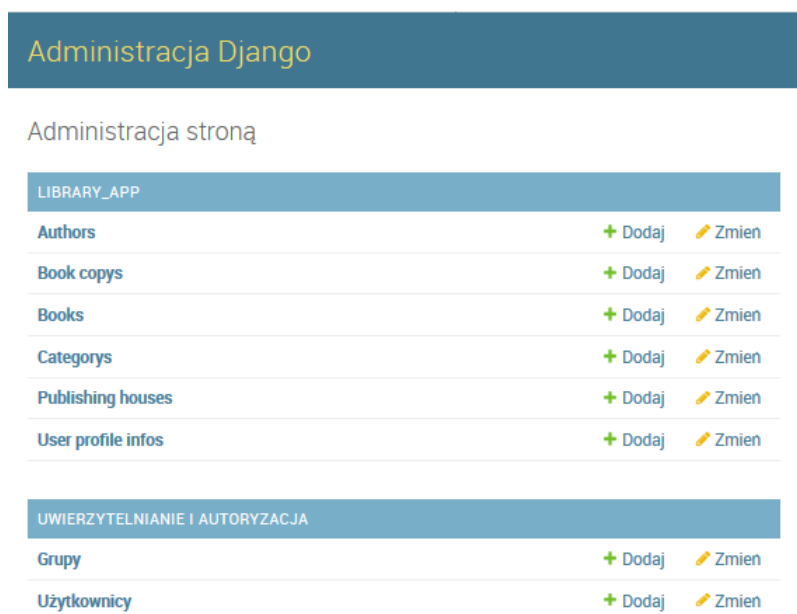
Dostęp do panelu administracyjnego.

Framework Django udostępnia panel administratora, który może być dowolnie modyfikowany. Jedynie użytkownik o statusie bibliotekarza ma możliwość dostępu do tego panelu (*Rys. 6.28*).



Rysunek 6.28: Formularz logowania do panelu administracyjnego.

Strona zawierająca panel administracyjny pozwala na wykonywanie, także tutaj, wszystkich akcji opisanych poprzednio. Na stronie wyświetlane są tabele zaimplementowane w projekcie (*Rys. 6.29*). Możliwe jest dodawanie, edytowanie oraz usuwanie danych zgromadzonych w poszczególnych tabelach (*Rys. 6.30*).



LIBRARY_APP		
Authors	+ Dodaj	Zmien
Book cypys	+ Dodaj	Zmien
Books	+ Dodaj	Zmien
Categorys	+ Dodaj	Zmien
Publishing houses	+ Dodaj	Zmien
User profile infos	+ Dodaj	Zmien

UWIERZYTELNIANIE I AUTORYZACJA		
Grupy	+ Dodaj	Zmien
Użytkownicy	+ Dodaj	Zmien

Rysunek 6.29: Strona główna panelu administracyjnego zawiera odnośniki do poszczególnych tabel.

Administracja Django

WITAJ, TOMASZ. [POKAŻ STRONĘ](#) / [ZMIANA HASŁA](#) / [WYLOGUJ SIĘ](#)

Początek » Library_App » Books

Zaznacz book do zmiany

DODAJ BOOK +

Q

Szukaj

Akcja:

Usun wybrane books

Wykonaj

Zaznaczono 1 z 14

<input type="checkbox"/>	TYTUŁ	AUTHORS	SLUG	NUMER ISBN	KATEGORIA	NUMER WYDANIA	WYDAWNICTWO	DATA PUBLIKACJI
<input type="checkbox"/>	Dziady część 3	Mickiewicz	3-598-21531-5	3-598-21531-5	Dramat	6	PWN	20 listopada 2018
<input type="checkbox"/>	Czysta architektura. Struktura i design oprogramowania. Przewodnik dla profesjonalistów	Martin	3-598-21531-2	3-598-21531-2	Informatyka	1	Helion	28 listopada 2018
<input checked="" type="checkbox"/>	Czysty kod. Podręcznik dobrego programisty	Martin	3-598-21528-2	3-598-21528-2	Informatyka	2	Helion	10 czerwca 2014
<input type="checkbox"/>	Aplikacje internetowe z Django. Najlepsze receptury	Bendoraitis	3-598-21527-4	3-598-21527-4	Informatyka	2	Helion	18 lutego 2018
<input type="checkbox"/>	Hartowanie Linuksa we wrogich środowiskach sieciowych.	Rankin	3-598-21525-8	3-598-21525-8	Informatyka	1	Helion	10 czerwca 2018

Rysunek 6.30: Strona panelu administracyjnego. Widoczne dane, dotyczące książek lub czasopism udostępnianych w bibliotece.

7 Uwagi i wnioski.

7.1 Możliwości dalszej rozbudowy projektu.

Implementacja projektu może zostać rozszerzona o następujące funkcjonalności:

- Możliwość rekomendacji wypożyczeń.

Opierając się na historycznych danych, dotyczących zwróconych przez użytkownika wypożyczeń, możliwe byłoby utworzenie systemu rekomendacji kolejnych książek i czasopism. Dane historyczne zawierają między informacje dotyczące: kategorii wypożyczanych książek, autorów książek lub wydawnictw, których zasoby były wypożyczane. Możliwe byłoby wtedy polecanie czytelnikowi interesujących go treści, bazując na poprzednich wypożyczeniach.

- Funkcjonalność rezerwacji egzemplarzy udostępnianych przez bibliotekę.

Na potrzeby usprawnienia dostępu do interesujących użytkownika zasobów istnieje możliwość zaimplementowania systemu rezerwacji egzemplarzy. Polegałoby to na umożliwieniu użytkownikowi oznaczenia jako zarezerwowane, książki lub czasopisma, które jest już wypożyczone. W momencie zwrotu zasobu użytkownik byłby informowany o tym fakcie.

- System płatności online dla nałożonych kar.

System informuje bibliotekarzy i użytkownika o należnościach, jakie są na niego nakładane, w obecnym stanie dane te mają jedynie charakter informacyjny. Możliwość rozbudowy polegałaby w tym przypadku na udostępnieniu użytkownikowi systemu płatności na stronie. Mógłby on wtedy opłacić naliczoną należność bezpośrednio na stronie. Ostatnim krokiem w takim przypadku byłby tylko zwrot zaległych wypożyczeń do biblioteki.

- Logowanie za pomocą mediów społecznościowych.

Ostatnią możliwością rozbudowy istniejącej implementacji systemu jest umożliwienie użytkownikom logowania do systemu za pomocą mediów społecznościowych. Framework Django udostępnia odpowiednie moduły pozwalające na realizację tej funkcjonalności [4].

7.2 Podsumowanie.

System w swojej obecnej formie spełnia wszystkie założenia wynikające z tematu pracy. System pozwala na usprawnienie działania biblioteki poprzez: możliwość gromadzenia informacji o wypożyczeniach, dodawanie do systemu informacji o fizycznych zasobach biblioteki oraz rejestrację i obsługę kont użytkowników.

Bibliografia.

- [1] Django - oficjalna dokumentacja. <https://docs.djangoproject.com/pl/1.11/>. z dnia: 2018-12-28.
- [2] J. Forcier, P. Bissex, and W. Chun. *Python i Django. Programowanie aplikacji webowych*. Helion, 2009.
- [3] R. Gryczan. *Bootstrap. Tworzenie własnych stylów graficznych*. Helion, 2017.
- [4] A. Mele. *Django. Praktyczne tworzenie aplikacji sieciowych*. Helion, 2016.
- [5] H. Percival. *TDD w praktyce. Niezawodny kod w języku Python*. Helion, 2015.
- [6] S. Rahman. *Bootstrap. Tworzenie interfejsów stron WWW. Technologia na start!* Helion, 2015.