

System obsługi biblioteki stworzony w ramach pracy inżynierskiej pod tytułem:
**“Projekt i implementacja systemu obsługi biblioteki w oparciu o technologie Django,
Bootstrap, HTML, CSS, JavaScript, SQL.”**

Autor: Wojciech Klusek

INSTRUKCJA INSTALACJI

Płyta zawiera plik „requirements.txt”, folder „library_project”, plik pracy inżynierskiej oraz niniejszą instrukcję.

Instalacja na systemie Linux

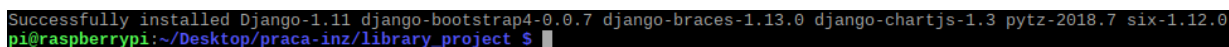
Wymagania wstępne:

- Zainstalowany pakiet python3 w wersji $\geq 3.5.3$,
- Zainstalowany pakiet pip3 w wersji $\geq 9.0.1$

Informacja: Podczas instalacji pakietu python3, zazwyczaj instalowany jest także pakiet pip3.

- Uruchomić konsolę
- Będąc w katalogu w którym znajduje się plik „requirements.txt” zainstalować niezbędne pakiety wpisując polecenie:

pip3 install -r requirements.txt



```
Successfully installed Django-1.11 django-bootstrap4-0.0.7 django-braces-1.13.0 django-chartjs-1.3 pytz-2018.7 six-1.12.0
pi@raspberrypi:~/Desktop/praca-inz/library_project $
```

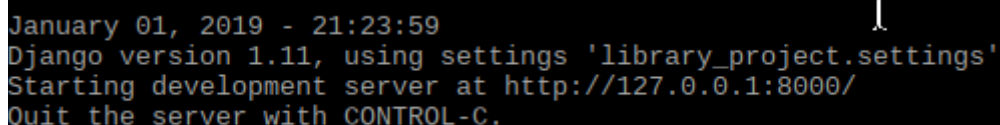
Rysunek 1. Informacja o prawidłowej instalacji wymaganych pakietów.

- Po instalacji pakietów pozostając w tym samym folderze przejść do folderu „library_project” poleceniem:

cd library_project/

- Uruchomić serwer oraz aplikację poleceniem:

python3 manage.py runserver



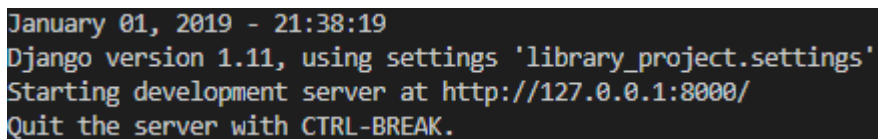
```
January 01, 2019 - 21:23:59
Django version 1.11, using settings 'library_project.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CONTROL-C.
```

Rysunek 2. Pomyślne wykonanie polecenia powinno wyświetlić następujące informacje.

- Po prawidłowym starcie serwera przejść na adres: <http://127.0.0.1:8000/>

Instalacja na systemie Windows

- Należy pobrać najnowszą wersję Python 3 ze strony: <https://www.python.org/downloads/> oraz zainstalować ją. W instalatorze zaznaczyć opcje „Add Python 3 to PATH”.
- Uruchomić konsolę cmd.
- Sprawdzić czy zainstalowana wersja Pythona ≥ 3.6 komendą:
python -V
- Sprawdzić wersję narzędzia pip komendą:
pip -V,
jeśli wersja < 18.0 należy zaktualizować narzędzie pip poleceniem
python -m pip install --upgrade pip
- Będąc w katalogu w którym znajduje się plik „requirements.txt” zainstalować niezbędne pakiety wpisując polecenie:
pip install -r requirements.txt
- Po instalacji pakietów pozostając w tym samym folderze przejść do folderu „library_project” poleceniem:
cd library_project/
- Uruchomić serwer oraz aplikację poleceniem:
python manage.py runserver



```
January 01, 2019 - 21:38:19
Django version 1.11, using settings 'library_project.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

Rysunek 4. Pomyślne wykonanie polecenia powinno wyświetlić następujące informacje.

- Po prawidłowym starcie serwera przejść na adres: <http://127.0.0.1:8000/>

Dane logowania

W projekcie istnieje użytkownik o statusie bibliotekarza:

- login: superuser
hasło: haslo123

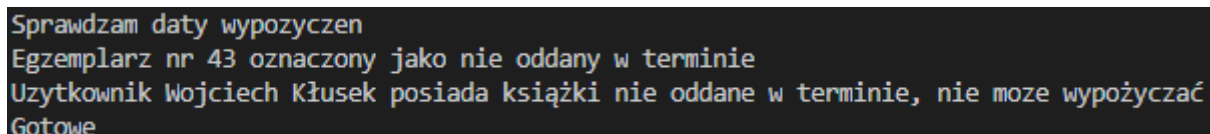
oraz użytkownik o statusie czytelnika:

- login: wojtek
hasło: haslo123

Użytkownicy posiadają wypożyczone zasoby biblioteki, baza danych zawiera informacje o przykładowych książkach i czasopismach.

Uwaga: W aplikacji zastosowano funkcjonalność oznaczania wypożyczeń jako „nieoddane w terminie”. W środowisku produkcyjnym byłby ona wywoływana specjalnym skrypcem poprzez cron, Możliwe jest jednak uruchomienie jej, z poziomu katalogu library_project/, poleceniem: **python manage.py check_borrows** – w systemie Windows lub, **python3 manage.py check_borrows** – w systemie Linux.

Skrypt ten wykryje wypożyczenia z przekroczoną datą zwrotu, odpowiednio je oznaczy, a także zablokuje użytkownikowi którego dotyczyło wypożyczenie, możliwość dalszego wypożyczania mu zasobów przez bibliotekarza.



```
Sprawdzam daty wypozycczen  
Egzemplarz nr 43 oznaczony jako nie oddany w terminie  
Uzytkownik Wojciech Kłusek posiada książki nie oddane w terminie, nie moze wypożyczać  
Gotowe
```

Rysunek 5. Przykład działania skryptu.

Cały system został także wystawiony na zewnętrznym hostingu i powinien być dostępny pod adresem: <https://wojtek9502.pythonanywhere.com/>

PythonAnywhere jest serwisem pozwalającym na hosting aplikacji napisanych przy użyciu frameworka Django.