
petycja-norweskie Documentation

Wydanie 0.1

Adam Dobrawy

01 sie 2017

1	Koncepcja aplikacji	3
1.1	Użycie i zasada działania	3
2	Instalacja	5
3	Wdrożenie	7
4	Testy	9
5	Kampanie (campaigns)	11
5.1	Założenia	11
5.2	Dodawanie kampanii	11
5.3	Architektura	12
6	Menu (menu)	13
6.1	Założenia	13
6.2	Architektura	13
7	Petycje (petitions)	15
7.1	Założenia	15
7.2	Architektura	15
8	Szablony (themes)	19
8.1	Założenia	19
8.2	Dostępne szablony	19
8.3	Dodawanie szablonu	19
8.4	Architektura	21
9	Indices and tables	23
	Indeks modułów pythona	25

Contents:

Koncepcja aplikacji

Petycje są formą bezpośredniego uczestnictwa obywateli w procesie sprawowania władzy. Zostały przewidziane w art. 63 Konstytucji RP:

Każdy ma prawo składać petycje, wnioski i skargi w interesie publicznym, własnym lub innej osoby za jej zgodą do organów władzy publicznej oraz do organizacji i instytucji społecznych w związku z wykonywanymi przez nie zadaniami zleconymi z zakresu administracji publicznej. Tryb rozpatrywania petycji, wniosków i skarg określa ustawa.

Przedstawiony system został zrealizowany w celu zapewnienia możliwości gromadzenia podpisów pod takimi petycjami w postaci elektronicznej w sposób efektywny i atrakcyjny dla użytkownika, a tym samym skuteczny.

System został opracowany początkowo na potrzeby kampanii norweskie.org. Jednak opierał się na wcześniejszych doświadczeniach Stowarzyszenia Sieć Obywatelska - Watchdog Polska między innymi z takimi kampaniami jak:

- NaszRzecznik.pl - kampania uruchomiona w 2015 roku celem wsparcia wyboru dr Adama Bodnara na funkcję Rzecznika Praw Obywatelskich - zakończona powodzeniem, w 2016 roku przekształcona w kampanie na rzecz obrony Rzecznika Praw Obywatelskich przed odwołaniem - zakończona powodzeniem,
- Jawna kampania wyborcza (wyłączona) - kampania uruchomiona w 2015 roku w związku z prezydencką kampanią wyborczą poświęconą presji na opublikowanie jeszcze przed wyborami prezydenckimi aktualizowanej informacji o otrzymanych przez partię w trakcie kampanii wyborczej darowiznach od osób fizycznych, która wywołała szeroką dyskusję o jawności kampanii wyborczej,
- Poprawka dla Fundacja Akademia Organizacji Obywatelskich (wyłączona) - kampania przeprowadzona w 2015 roku na przeciwdziałaniu ustawowej poprawce o zakazie finansowania „promocji 1%” ze środków uzyskanych w ramach „mechanizmu 1%” - zakończona sukcesem,

1.1 Użycie i zasada działania

Korzystanie z aplikacji warto rozpocząć od stworzenia dedykowanego szablonu (zob. *[Dodawanie szablonu](#)*). Następnie dodania kampanii (zob. *[Dodawanie kampanii](#)*), utworzenia petycji (zob. *[Petycje \(petitions\)](#)*). Zadaniem użytkowników strony będzie utworzenie podpisów (zob. *[petycja_norweskie.petitions.models.Signature](#)*).

Aplikacja została skonstruowana wokół kampanii (zob. *Kampanie (campaigns)*), która grupuje jedną lub więcej petycji (zob. *Petycje (petitions)*). Poszczególne petycje mają określony formularz, pola w nim, a także wymagane zgody. To w relacji z petycją znajdują się wszelkie składane podpisy.

ROZDZIAŁ 2

Instalacja

Został opracowany playbook Ansible, który zapewnia uruchomienie aplikacji w środowisku developerskim. Przedstawia on także podstawowe kroki, które są konieczne do uruchomienia aplikacji. Dostępny jest on w pliku `vagrant_provision_ansible.yaml`. Zalecane jest wykorzystanie przedstawionego playbooka wraz z środowiskiem wirtualizacyjnym Vagrant. Wówczas konfiguracja całego środowiska to:

```
$ vagrant up --provision
$ vagrant ssh -
vagrant@vagrant:/vagrant$ python manage.py runserver 0.0.0.0:8000
```

Następnie można przejść w przeglądarce pod adres `http://localhost:8000`.

Po zakończeniu pracy można wykonać w celu skasowania wirtualnej maszyny:

```
$ vagrant destroy
```

Alternatywnie w celu zaoszczędzenia pamięci RAM można ją wyłączyć uśpić:

```
$ vagrant suspend
```

Warto także zwrócić uwagę na polecenie zapewniające utworzenie użytkownika administracyjnego:

```
vagrant@vagrant:/vagrant$ python manage.py createsuperuser
```

Jeżeli zepsujesz sobie bazę danych wykonaj:

```
vagrant@vagrant:/vagrant$ sudo -H mysql 'drop database petycja_norweskie';
$ vagrant provision
```

Jeżeli chcesz skonfigurować maszynę od nowa wykonaj:

```
$ vagrant destroy -f && vagrant up --provision
```

Jeżeli chcesz upewnić się co do aktualności konfiguracji możesz wykonać:

```
$ vagrant provision
```

Wdrożenie

Wdrożenie aplikacji wymaga:

- dostępności Python>3.3 i menadżera pakietów `pip`,
- skonfigurowania serwera baz danych MariaDB,
- wywołania serwera aplikacyjnego `gunicorn`,
- skonfigurowania serwera Nginx jako reverse-proxy w `Gunicorn`.

Ponadto konieczne jest ustawienie następujących zmiennych środowiskowych:

- `DJANGO_SECRET_KEY` - losowa i poufność wartość zgodnie z dokumentacją Django dla `SECRET_KEY`,
- `DJANGO_SETTINGS_MODULE` o wartości `config.settings.production` dla określenie pliku konfiguracyjnego wykorzystanego po załadowaniu aplikacji,
- `DJANGO_SENTRY_DSN` - adres `Sentry DSN` służący do wskazania narzędzia monitoringu wyjątków,
- `DJANGO_ADMIN_URL`` o wartości np. „admin/” dla określenia ścieżki panelu administracyjnego,
- `CACHE_URL` o wartości zgodnej z `django-environ` dla ustawienia mechanizmu cache.

W środowisku Stowarzyszenie wdrożenie odbywa się z wykorzystaniem roli `Ansible watchdogpolska.django`.

Testy automatyczne zostały oparte o wbudowane w Django mechanizmy. Dostał wykorzystany także domyślny „test runner”. Automatyzację testów różnych konfiguracji np. wersji zależności zapewnia `tox`.

Aby wypisać dostępne środowiska należy wykonać:

```
tox -l
```

Wówczas możliwy jest wybór środowiska testów i wykonanie:

```
tox -e dj111-coveralls
```

Zostały wdrożone ciągle testy integracyjne z wykorzystaniem TravisCI.

Kampanie (campaigns)

5.1 Założenia

Moduł zapewnia mechanizm budowania wielu kampanii w ramach indywidualnej instancji aplikacji. Pozwala to ograniczyć liczbę aplikacji, które będą uruchomione dla różnych kampanii. Dla uruchomienia dodatkowej instancji aplikacji konieczne jest zagwarantowanie, że

Każda kampania stanowi jedną lub więcej petycji (*Petycje (petitions)*).

Każda kampania może mieć skonfigurowany indywidualny wystrój, dzięki mechanizmowi szablonów (*Szablony (themes)*).

Mechanizm kampanii odpowiedzialny jest także za kontrolę dostępu. Umożliwia bowiem określenia użytkowników, którzy bez uprawnień administracyjnych mają możliwość zarządzania ankietami.

5.2 Dodawanie kampanii

Uruchomienie kampanii wymaga podjęcia następujących kroków:

1. aktualizacja serwera WWW do obsługi nowej domeny
2. aktualizacja dopuszczalnych adresów domenowych - zob. `ALLOWED_HOSTS` poprzez zmienną środowiskową `DJANGO_ALLOWED_HOSTS`
3. dodawania nowej strony - zob. `django.contrib.sites`
4. dodania kampanii w panelu administracyjnym - zob. `petycja_norweskie.campaigns.admin.CampaignAdmin`

5.3 Architektura

5.3.1 Model

```
class petycja_norweskie.campaigns.models.Campaign(id, created, modified, name, organizer, theme, site, site_title, site_subtitle, show_title)
```

Parametry

- **id** (*AutoField*) – Id
- **created** (*AutoCreatedField*) – Utworzony
- **modified** (*AutoLastModifiedField*) – Zmodyfikowany
- **name** (*CharField*) – Nazwa
- **organizer** (*CharField*) – Osoba lub organizacja odpowiedzialna za organizację kampanii
- **theme_id** (*ForeignKey* to *petycja_norweskie.themes.models.Theme*) – Theme
- **site_id** (*ForeignKey* to *django.contrib.sites.models.Site*) – Strona użyta w kampanii
- **site_title** (*CharField*) – Nazwa
- **site_subtitle** (*CharField*) – Podtytuł
- **show_title** (*BooleanField*) – Pokaż tytuł petycji

5.3.2 Panel administracyjny

```
class petycja_norweskie.campaigns.admin.CampaignAdmin(model, admin_site)  
Admin View for Campaign
```

5.3.3 Widoki

5.3.4 Fabryki

6.1 Założenia

Moduł zapewnia mechanizm menu nawigacyjnego dla poszczególnych stron internetowych. Odnośniki mogą być:

- odnośnikami wewnętrznymi, a wówczas weryfikowana jest zgodność adresów z aplikacją, choć to nie znaczy, że pod podanym adresem zawarta jest treść
- odnośnikami zewnętrznymi `https://` lub `http:`,
- odnośnikami e-mailowymi `mailto:`.

6.2 Architektura

6.2.1 Model

class `petycja_norweskie.menu.models.Element` (*id, created, modified, name, url, parent, visible, position*)

Parametry

- **id** (*AutoField*) – Id
- **created** (*AutoCreatedField*) – Utworzony
- **modified** (*AutoLastModifiedField*) – Zmodyfikowany
- **name** (*CharField*) – Nazwa
- **url** (*CharField*) – Url
- **parent_id** (*ForeignKey* to `petycja_norweskie.menu.models.Element`) – Rodzic
- **visible** (*BooleanField*) – Zaznacz, aby oznaczyć szablon jako publicznie widoczny

- **position** (*SmallIntegerField*) – Pozycja

6.2.2 Panel administracyjny

class `petycja_norweskie.menu.admin.ElementAdmin` (*model*, *admin_site*)
Admin View for Element

class `petycja_norweskie.menu.admin.ElementInline` (*parent_model*, *admin_site*)
Tabular Inline View for Element

model
alias klasy `Element`

6.2.3 Procesorzy kontekstu

`petycja_norweskie.menu.context_processors.menu` (*request*)
A context processor which provide menu in menu template variable.

Parametry **request** (*HttpRequest*) – A django standard request object

Example

Menu render is very simple and effective. For example:

```
{% if menu %}
<ul>
  {% for el in menu %}
  <li>
    <a href="{{el.url}}">{{el}}</a>
    {% if el.children_set %}
    <ul>
      {% for child in el.children_set %}
        <li><a href="{{child.url}}">{{child}}</a>
      {% endfor %}
    </ul>
    {% endif %}
  </li>
  {% endfor %}
</ul>
{% endif %}
```

6.2.4 Widoki

7.1 Założenia

Komponent zapewnia mechanizm petycji, a także gromadzenia pod nimi podpisów, co stanowi podstawowy cel funkcjonowania projektu. Każda petycja ma określoną kampanie (zob. [Menu \(menu\)](#)).

Mechanizm petycji jest szeroko konfigurowalny zapewniając możliwość ukrycia każdego z pól formularzy, co umożliwia uwzględnienie specyfiki kampanii.

Ponadto jest przewidziana funkcjonalność pól zgód. W celu ich skorzystania należy pierw zdefiniować definicję zgody dla danej petycji. Można przy tym określić czy zgoda będzie wymagana, czy opcjonalna, a także wzajemną kolejność zgód w formularzu. Zapewnia to szerokie możliwości uzyskania zgód zgodnie z różnorodnymi potrzebami i wymogami prawa.

Zgromadzone podpisy możliwe są do wyeksportowania w szeregu formatach. Eksport obejmuje także informacje o udzielonych zgodach.

7.2 Architektura

7.2.1 Model

```
class petycja_norweskie.petitions.models.Permission(id, definition, signature, value)
```

Parametry

- **id** (*AutoField*) – Id
- **definition_id** (*ForeignKey* to *petycja_norweskie.petitions.models.PermissionDefinition*) – Definicja zgody
- **signature_id** (*ForeignKey* to *petycja_norweskie.petitions.models.Signature*) – Podpis
- **value** (*BooleanField*) – Wartość

```
class petycja_norweskie.petitions.models.PermissionDefinition(id, petition, text,
                                                             default, required,
                                                             ordering)
```

Parametry

- **id** (*AutoField*) – Id
- **petition_id** (*ForeignKey* to *petycja_norweskie.petitions.models.Petition*) – Petycja
- **text** (*TextField*) – Text
- **default** (*BooleanField*) – Czy domyślnie zaznaczone?
- **required** (*BooleanField*) – Czy wymagane?
- **ordering** (*PositiveSmallIntegerField*) – Określ kolejność

```
class petycja_norweskie.petitions.models.Petition(id, created, modified, campaign,
                                                    name, slug, title, text,
                                                    overview, finish_message,
                                                    ask_first_name, ask_second_name,
                                                    ask_organization, ask_city,
                                                    ask_email, first_name_label,
                                                    second_name_label, organization_label,
                                                    city_label, email_label,
                                                    sign_button_text, paginate_by,
                                                    is_published, is_active, front,
                                                    disabled_warning, disabled_message)
```

Parametry

- **id** (*AutoField*) – Id
- **created** (*AutoCreatedField*) – Utworzony
- **modified** (*AutoLastModifiedField*) – Zmodyfikowany
- **campaign_id** (*ForeignKey* to *petycja_norweskie.campaigns.models.Campaign*) – Kampania
- **name** (*CharField*) – Nazwa
- **slug** (*CharField*) – Zmodyfikuj, aby zmienić adres petycji
- **title** (*CharField*) – Tytuł
- **text** (*TextField*) – Treść
- **overview** (*TextField*) – Zwarte omówienie przedmiotu petycji zachęcające do podpisania petycji.
- **finish_message** (*TextField*) – Wiadomość pokazywana po podpisach
- **ask_first_name** (*BooleanField*) – Zapytaj o imię
- **ask_second_name** (*BooleanField*) – Zapytaj o nazwisko
- **ask_organization** (*BooleanField*) – Zapytaj o organizację
- **ask_city** (*BooleanField*) – Zapytaj o miasto
- **ask_email** (*BooleanField*) – Zapytaj o e-mail
- **first_name_label** (*CharField*) – Etykieta pola imienia

- **second_name_label** (*CharField*) – Etykieta pola nazwiska
- **organization_label** (*CharField*) – Etykieta pola nazwy organizacji
- **city_label** (*CharField*) – Etykieta pola miejscowości
- **email_label** (*CharField*) – Etykieta pola adresu e-mail
- **sign_button_text** (*CharField*) – Etykieta przycisku zgłoszeniowego
- **paginate_by** (*SmallIntegerField*) – Określa liczbę podpisów na jednej podstronie z listą podpisów
- **is_published** (*BooleanField*) – Opublikowane na stronie?
- **is_active** (*BooleanField*) – Otwarte na nowe podpisy?
- **front** (*BooleanField*) – Tutaj powinna być jedna dostępna strona
- **disabled_warning** (*TextField*) – Wiadomość, gdy ktoś próbuje wprowadzić podpis, pomimo wyłączenia formularza.
- **disabled_message** (*TextField*) – Wiadomość zamieszczona na stronie, jeśli możliwość składania podpisów została wyłączona.

```
class petycja_norweskie.petitions.models.Signature(id, created, modified, petition,
                                                    first_name, second_name, organization, city, email, counter)
```

Parametry

- **id** (*AutoField*) – Id
- **created** (*AutoCreatedField*) – Utworzony
- **modified** (*AutoLastModifiedField*) – Zmodyfikowany
- **petition_id** (*ForeignKey* to *petycja_norweskie.petitions.models.Petition*) – Petycja
- **first_name** (*CharField*) – Imię
- **second_name** (*CharField*) – Nazwisko
- **organization** (*CharField*) – Organizacja
- **city** (*CharField*) – Miejscowość
- **email** (*EmailField*) – Adres e-mail
- **counter** (*SmallIntegerField*) – Numer

7.2.2 Formularze

7.2.3 Panel administracyjny

7.2.4 Widoki

7.2.5 Fabryki

8.1 Założenia

Komponent określa szablony i ich parametry, które są możliwe podczas konfiguracji kampanii.

Szczególną rolę pełni pole `Theme.prefix`, gdyż określa dopuszczalne wartości.

Każdy szablon winien zapewniać następujące pliki szablonów Django:

```
$ find $(find -type d -name 'norweskie') -type f
./petycja_norweskie/templates/norweskie/base.html
./petycja_norweskie/petitions/templates/petitions/norweskie/field_custom.html
./petycja_norweskie/petitions/templates/petitions/norweskie/signature_form.html
./petycja_norweskie/petitions/templates/petitions/norweskie/signature_list.html
./petycja_norweskie/petitions/templates/petitions/norweskie/petition_success.html
./petycja_norweskie/petitions/templates/petitions/norweskie/petition_detail.html
```

W wskazanym przykładzie «norweskie» stanowi odpowiedni prefix.

8.2 Dostępne szablony

8.2.1 norweskie.org

8.3 Dodawanie szablonu

W przypadku dodania nowego szablonu należy wykonać:

```
python manage.py makemigrations themes --empty --name "add theme {{prefix}}"
```

W nowoutworzonym pliku wstawić treść zgodnie z szablonem:



Fundusze Mechanizmu Finansowego EOG dla organizacji

PODPISZ PODPISALI KONTAKT

niezależne od rządu

Szanowni Państwo,
Koleżanki i Koledzy,
Zapraszamy do podpisania listu, w którym protestujemy przeciwko zorganizowanej akcji, której celem jest przejęcie przez polską administrację państwową zarządzania środkami EOG przeznaczonymi na wsparcie rozwoju społeczeństwa obywatelskiego.

W liście apelujemy do rządu Królestwa Norwegii, z którego budżetu pochodzi największa część środków, o pozostawienie zarządzania tymi funduszami w rękach niezależnego operatora, wyłonionego w otwartej procedurze konkursowej. Sprzeciwiamy się także pozabawionym podstaw merytorycznych zarzutom kierowanym pod adresem Fundacji Batorego – obecnego operatora funduszy.

Na podpisy czekamy do czwartku 20 kwietnia br. Po tym terminie list zostanie przekazany Ambasadzie Norwegii oraz do wiadomości Pełnomocnikowi Rządu do Spraw Społeczeństwa Obywatelskiego i Równego Traktowania.

Komitet Organizacyjny
11 kwietnia 2017 roku

List otwarty organizacji pozarządowych w sprawie środków Mechanizmu Finansowego EOG

Stanowczo protestujemy przeciw prowadzonej ostatnio, zorganizowanej akcji rządu z udziałem publicznych mediów i grupy organizacji pozarządowych, mającej na celu przejęcie przez administrację państwową zarządzania środkami Mechanizmu Finansowego Europejskiego Obszaru Gospodarczego, przeznaczonymi na wsparcie rozwoju społeczeństwa obywatelskiego.

Uważamy, że tylko niezależność i oparty na doświadczeniu profesjonalizm operatora konkursów grantowych gwarantuje efektywne, zgodne z przeznaczeniem wykorzystanie tych środków. Obawiamy się, że ani nieistniejące jeszcze, budzące już teraz wiele kontrowersji, Narodowe Centrum Rozwoju Społeczeństwa Obywatelskiego, ani żadna inna instytucja państwowa nie jest w stanie spełnić takich wymagań. Ostatnie przykłady nieoczekiwanych zmian zasad, dokumentów, wzorów umów, jakie miały miejsce w trakcie konkursów grantowych dla organizacji pozarządowych, dowodzą słabości – i tak wcześniej słabych – standardów grantodawczych administracji państwowej. Obawiamy się również ryzyka wykorzystania funduszy EOG do realizacji partykularnych celów polityki rządu, o czym świadczą mogą ostatnie wypowiedzi jego członków.

Wyrażamy sprzeciw wobec bezpardonowych ataków na Fundację Batorego – dotychczasowego operatora funduszy EOG – prowadzonych przez przedstawicieli rządu, części mediów i grupy organizacji pozarządowych. Ich głównym celem jest dyskredytacja dorobku jednej z najważniejszych niezależnych instytucji grantodawczych w kraju tak, aby osłabić pozycję fundacji jako potencjalnego operatora kolejnej transzy środków Mechanizmu Finansowego EOG, zwiększając jednocześnie szanse rządu na przejęcie tej funkcji.

Zarzuty formułowane są w tendencyjny sposób, w oparciu o nieprawdziwe lub manipulowane informacje – np. mówią o przekazaniu 150 mln zł organizacjom liberalno-lewicowym (co przekracza całkowitą kwotę dotacji przekazanych wszystkim organizacjom) lub o „zawrotnych” środkach przekazanych z kolei aktywistom LGBT (wartość wsparcia finansowego udzielonego takim organizacjom to 2% łącznej kwoty wypłaconych dotacji). Podobnie wadliwie konstruowanych zarzutów jest więcej – nie zgadzamy się z nimi.

Głęboko niepokoi nas również fakt, iż w opisywane działania włączono organizacje pozarządowe. Być może niektóre z nich dały się przekonać złudnymi obietnicami, iż rząd może wydatkować fundusze EOG na określone przez siebie cele, być może inne uwierzyły w nierzetelnie prezentowane dane o jednym z najlepiej realizowanych w ostatnich latach w Polsce konkursów grantowych. W efekcie tworzone są kolejne podziały w społeczeństwie, tym razem w środowisku, które dotąd – mimo różnic – umiało ze sobą rozmawiać i współpracować. Nie tworzy to dobrego klimatu dla wzmocnienia sektora obywatelskiego.

Wyraż poparcie

Nazwa organizacji

E-mail do organizacji

PODPISZ LIST

PODPISALI

List podpisało 276 organizacji. Dołącz do nich!




```
# -*- coding: utf-8 -*-
from __future__ import unicode_literals

from django.db import migrations

def update_theme_forward(apps, schema_editor):
    Theme = apps.get_model('themes', 'Theme')
    Theme.objects.update_or_create(
        prefix="norweskie",
        defaults={
            'name': 'Niezależne fundusze norweskie',
            'description': '',
            'authorship': 'Kamil Breguła (mik-laj)'
        }
    )

class Migration(migrations.Migration):
    dependencies = [
        ('themes', '0001_initial'),
    ]

    operations =
    migrations.RunPython(update_theme_forward)
]
```

Warto także w niniejszej dokumentacji zamieścić podgląd.

8.4 Architektura

8.4.1 Model

class `petycja_norweskie.themes.models.Theme` (*id, created, modified, name, description, authorship, prefix*)

Parametry

- **id** (*AutoField*) – Id
- **created** (*AutoCreatedField*) – Utworzony
- **modified** (*AutoLastModifiedField*) – Zmodyfikowany
- **name** (*CharField*) – Nazwa
- **description** (*TextField*) – Nazwa
- **authorship** (*CharField*) – Autorstwo
- **prefix** (*CharField*) – Systemowa nazwa prefiksu szablonu

8.4.2 Panel administracyjny

8.4.3 Widoki

Indices and tables

- `genindex`
- `modindex`
- `search`

p

`petycja_norweskie.campaigns.admin`, [12](#)
`petycja_norweskie.campaigns.models`, [12](#)
`petycja_norweskie.campaigns.tests.factories`,
[12](#)
`petycja_norweskie.campaigns.views`, [12](#)
`petycja_norweskie.menu.admin`, [14](#)
`petycja_norweskie.menu.context_processors`,
[14](#)
`petycja_norweskie.menu.models`, [13](#)
`petycja_norweskie.menu.views`, [14](#)
`petycja_norweskie.petitions.admin`, [17](#)
`petycja_norweskie.petitions.forms`, [17](#)
`petycja_norweskie.petitions.models`, [15](#)
`petycja_norweskie.petitions.tests.factories`,
[17](#)
`petycja_norweskie.petitions.views`, [17](#)
`petycja_norweskie.themes.admin`, [21](#)
`petycja_norweskie.themes.models`, [21](#)
`petycja_norweskie.themes.views`, [21](#)

C

Campaign (klasa w module
cja_norweskie.campaigns.models), 12

CampaignAdmin (klasa w module
cja_norweskie.campaigns.admin), 12

E

Element (klasa w module
cja_norweskie.menu.models), 13

ElementAdmin (klasa w module
cja_norweskie.menu.admin), 14

ElementInline (klasa w module
cja_norweskie.menu.admin), 14

M

menu() (w module pety-
cja_norweskie.menu.context_processors),
14

model (atrybut petycja_norweskie.menu.admin.ElementInline),
14

P

Permission (klasa w module pety-
cja_norweskie.petitions.models), 15

PermissionDefinition (klasa w module pety-
cja_norweskie.petitions.models), 15

Petition (klasa w module pety-
cja_norweskie.petitions.models), 16

petycja_norweskie.campaigns.admin (moduł), 12

petycja_norweskie.campaigns.models (moduł), 12

petycja_norweskie.campaigns.tests.factories (moduł), 12

petycja_norweskie.campaigns.views (moduł), 12

petycja_norweskie.menu.admin (moduł), 14

petycja_norweskie.menu.context_processors (moduł), 14

petycja_norweskie.menu.models (moduł), 13

petycja_norweskie.menu.views (moduł), 14

petycja_norweskie.petitions.admin (moduł), 17

petycja_norweskie.petitions.forms (moduł), 17

petycja_norweskie.petitions.models (moduł), 15

petycja_norweskie.petitions.tests.factories (moduł), 17

petycja_norweskie.petitions.views (moduł), 17

petycja_norweskie.themes.admin (moduł), 21

petycja_norweskie.themes.models (moduł), 21

petycja_norweskie.themes.views (moduł), 21

S

Signature (klasa w module pety-
cja_norweskie.petitions.models), 17

T

Theme (klasa w module pety-
cja_norweskie.themes.models), 21