

# Laboratorium nr 5

## Uwierzytelnianie z użyciem Devise

dr inż. Aleksander Smywiński-Pohl

Elektroniczne Przetwarzanie Informacji  
Konsultacje: czw. 14.00-15.30, pokój 3.211

# Plan prezentacji

Wstęp

Konfiguracja

Użycie modelu User

# Wprowadzenie

- ▶ uwierzytelnianie, autentykacja – zweryfikowanie, że osoba jest tym, za kogo się podaje
- ▶ autoryzacja, kontrola dostępu – zweryfikowanie, że osoba wykonująca określoną operacją, ma do tego prawo

Uwierzytelnianie:

- ▶ Devise
- ▶ Authlogic
- ▶ restful-authentication

Autoryzacja:

- ▶ cancan
- ▶ declarative\_authorized
- ▶ acl9

# Devise

- ▶ najbardziej popularny
- ▶ wysoce konfigurowalny
- ▶ działa z Railsami v. 4
- ▶ spore możliwości, np.
  - ▶ trwała sesja
  - ▶ resetowanie hasła przez e-mail
  - ▶ różne algorytmy hashowania

## Dostępne moduły (1)

- ▶ **confirmable** – wymagane jest potwierdzenia konta (e-mail)
- ▶ **lockable** – umożliwia blokowania konta po nieudanych próbach logowania
- ▶ **timeoutable** – inwaliduje sesje, po określonym czasie bezczynności
- ▶ **omniauthable** – dodaje wsparcie dla Omniauth (np. logowanie z użyciem Facebooka, Githuba, Twittera, Google+, itp.)
- ▶ **database\_authenticatable** – dane uwierzytelniające zapisywane są w bazie danych

## Dostępne moduły (2)

- ▶ **registerable** – umożliwia rejestrację za pomocą standardowego formularza
- ▶ **recoverable** – umożliwia odzyskanie hasła
- ▶ **rememberable** – umożliwia zapamiętanie sesji na danym komputerze
- ▶ **trackable** – rejestruje informacje o próbach logowania
- ▶ **validatable** – (domyślnie) weryfikuje poprawność wprowadzonego hasła

# Plan prezentacji

Wstęp

Konfiguracja

Użycie modelu User

# Konfiguracja Devise

1. W pliku Gemfile dodajemy linijkę  
gem 'devise'
2. ▷ bundle install
3. ▷ rails generate devise:install
4. ▷ rails generate devise user
5. ▷ rake db:migrate
6. Restartujemy serwer



# Model User

- ▶ jest zwykłym modelem railsowym, który może być modyfikowany przez programist(k)ę
- ▶ dziedziczy z ActiveRecord::Base
- ▶ domyślnie posiada pola: e-mail (używany jako login) i hasło

```
class User < ActiveRecord::Base
  # Include default devise modules. Others available are:
  # :confirmable, :lockable, :timeoutable and :omniauthable
  devise :database_authenticatable, :registerable,
         :recoverable, :rememberable, :trackable, :validatable
end
```

## Kontroler – zastosowania uwierzytelniania

- ▶ filtr `authenticate_user!` sprawdza, czy użytkownik jest zalogowany i przekierowuje na stronę logowania, jeśli nie jest zalogowany
- ▶ przykład: `app/controllers/authors_controller.rb`

```
class AuthorsController < ApplicationController
  before_action :set_author, only: [:show, :edit, :update, :destroy]
  # nowa linijka
  before_action :authenticate_user!, except: [:index, :show]
```
- ▶ zaglądamy pod adres  
`http://adres.hosta.pl:port/authors`
- ▶ próbujemy wykonać akcję `New author`
- ▶ rejestrujemy się
- ▶ stworzymy nowego autora

# Filtry w kontrolerze

- ▶ filtr `before_action` wykonuje akcję dodatkową przed wykonaniem akcji głównej
- ▶ `only`: ogranicza ten filtr tylko do wskazanych akcji
- ▶ `except`: określa akcje, do których filtr nie jest stosowany – lepsze rozwiązanie w odniesieniu do uwierzytelniania, tzw. *future proof*

# Widok – zastosowanie uwierzytelniania

- ▶ helper `user_signed_in?` sprawdza, czy użytkownik jest zalogowany
- ▶ przykład: `app/views/layouts/application.html.erb`

```
<body>
<!-- nowy kod -->
<% if user_signed_in? %>
<%= link_to "logout", destroy_user_session_path, method: :delete %>
<% else %>
<%= link_to "login", new_user_session_path %>
<% end %>
<!-- koniec nowego kodu -->
<%= yield %>
</body>
```

# Tłumaczenie komunikatów

- ▶ Tworzymy plik `config/locales/devise.pl.yml`

```
pl:
  devise:
    sessions:
      signed_in: "Zostałeś(aś) zalogowany(a)."
```

```
      signed_out: "Zostałeś(aś) wylogowany(a)."
```

- ▶ W pliku `config/application.rb` odszukujemy linijkę

```
# config.i18n.default_locale = :de
```

odkomentujemy i zamieniamy na:

```
config.i18n.default_locale = :pl
```

- ▶ Zatrzymujemy serwer i ponownie go uruchamiamy.
- ▶ Zalogowujemy się i wylogowujemy.
- ▶ Gotowe tłumaczenia

<https://github.com/plataformatec/devise/wiki/I18n>

# Plan prezentacji

Wstęp

Konfiguracja

Użycie modelu User

# Kontroler EntriesController

▶ `app/controllers/entries_controller.rb`

```
class EntriesController < ApplicationController
  # nowa linijka
  before_action :authenticate_user!, except: [:index]

  # ...

  def new
    @entry = Entry.new
    # nowa linijka
    @entry.name = current_user.email
  end
end
```

▶ Dodajemy nowy wpis w książce adresowej.

# Zadania

- ▶ Zabezpiecz metody modyfikujące książek przed niezalogowanymi użytkownikami
- ▶ Dodaj link dla zalogowanych użytkowników pozwalający na zmianę hasła i adresu e-mail



# Materiały

- ▶ Devise <https://github.com/plataformatec/devise>

Dziękuję!