

Movie Pole publiczna filmoteka

Silesian Ruby Users' Group

Marek Kowalcze
Jakub Kuźma
Wojciech Wnętrzak

15 grudnia 2008

Sposób na nudę

```
10 PRINT "Otwórz http://imdb.com/ lub podobną stronę."  
20 PRINT "Szukaj dobrych filmów do obejrzenia."  
30 PRINT "Otwórz http://mininova.org/ lub podobną."  
40 PRINT "Odszukaj wybrany film."  
50 PRINT "Pobierz film używając wybranego programu p2p."  
60 PRINT "Obejrzyj film."  
70 GOTO 10
```

Publiczna filmoteka

- wykorzystanie serwisów: IMDb, Mininova, itp.
- aplikacja Ruby on Rails - baza filmów, torrentów, kanał RSS
- klient desktopowy - czytnik RSS

Skupmy się na Ruby on Rails

- baza danych: filmy, torrenty
- dane dodawane z zewnątrz
- lista filmów
- kanał RSS

Do dzieła!

- generujemy aplikację
- omówienie struktury
- skrypty (serwer, konsola, generatory)
- zadania Rake

Model Movie

- generowanie modelu
- tworzenie migracji (pola: tytuł, identyfikator IMDb, liczba głosów)
- dodanie podstawowych walidacji (tytuł, identyfikator IMDb)

Kontroler Movies

- generowanie kontrolera
- tworzenie akcji index (lista filmów)
- tworzenie akcji show (szczegóły filmu)
- tworzenie ścieżek (routing)

Widoki: index, show

- tworzenie szablonów ERb
- nawigacja pomiędzy widokami
- layout

Torrenty

- tworzenie modelu i migracji
- walidacje (rozmiar pliku)
- dodanie asocjacji z filmami
- uzupełnianie widoków

Posprzątajmy trochę!

- podział widoku (partial “torrent”)
- utworzenie metody pomocniczej `imdb_url`
- utworzenie metody wypisującej rozmiar `in_megabytes`

Udostępnienie kanału RSS

- tworzenie widoku

IMDb - problemy

- brak publicznego API
- konieczność parsowania HTML
- zwracanie danych w przystępnej postaci

Co chcemy uzyskać?

```
imdb = IMDB::NowPlaying.new  
results = imdb.movies(2008, 12)
```

```
[{ :imdb_id => "0441773", :title => "Kung Fu Panda" },  
 { :imdb_id => "0493464", :title => "Wanted" }]
```

Co chcemy uzyskać? c.d.

```
imdb = IMDB::FullInformation.new  
imdb.full_information("0441773")
```

```
{ :title => "Kung Fu Panda",  
  :rating => 7.8,  
  :votes => 41106,  
  :genres => ["Comedy", "Animation", "Action"],  
  :directors => ["Mark Osborne", "John Stevenson"] }
```

IMDB::NowPlaying

```
module IMDB
  class NowPlaying
    def movies(year, month)
      uri = "#{BASE_URI}/nowplaying/%04d/%02d" % [year, month]
      doc = Hpricot(open(uri, HEADERS))
      results = []
      (doc / "a.title").each do |a|
        result = {}
        match = a["href"].match(/\/title\/tt(\d+)/)
        result[:imdb_id] = match[1] if match
        result[:title] = a.inner_html
        results << result unless result.empty?
      end
      results
    end
  end
end
```

Integracja aplikacji z backendami

- skrypt runner
- uruchamiany z zewnątrz (cron)

Aplikacja wxRuby

- wykorzystanie dowiązań do wxWidgets
- proste GUI
- wyświetlanie zawartości kanału RSS (FeedNormalizer)

Interfejs - wxGlade

- wzorowany na Glade (GTK+)
- „klikane” tworzenie interfejsu
- eksport do XRC (XML Resource)

Główne okno aplikacji

```
class MoviePoleMain < Wx::Frame
  def initialize
    super
    Wx::XmlResource.get.load_frame_subclass(self, nil, "moviepoleframe")

    @feed_list = find_window_by_id(Wx::xrcid("feedlist"))
    @feed_content = find_window_by_id(Wx::xrcid("feed"))
    @close = find_window_by_id(Wx::xrcid("close"))

    @feed = FeedNormalizer::FeedNormalizer.parse(open("http://localhost:3000/movies.rss"))
    @feed_list.set(@feed.entries.map(&:title))

    evt_button @close.id, lambda { close }
    evt_listbox @feed_list.id, :on_feed_list
  end

  def on_feed_list
    selected = @feed_list.selections.first
    @feed_content.page = @feed.entries[selected].content
  end
end
```

Uruchamianie aplikacji

```
require "rubygems"  
require "wx"  
require "feed-normalizer"  
require "open-uri"  
  
require "movie_pole_main"  
  
Wx::App.run do  
  Wx::XmlResource.get.load("movie_pole.xrc")  
  self.app_name = "Movie Pole"  
  frame = MoviePoleMain.new  
  frame.show  
end
```

Ruby in Shoes

- proste tworzenie kolorowych aplikacji
- wieloplatformowość
- zintegrowane środowisko uruchomieniowe

Na zakończenie

- http://github.com/ncr/movie_pole
- <http://www.srug.pl/>
- spotkania@srug.pl
- <http://trix.lighthouseapp.com/projects/20503-movie-pole>

Dziękujemy za wytrwałość!