

## Web Applications in Ruby on Rails (code: Rails)

### Overview

— Ruby on Rails\* is an advanced environment (framework) used for creating attractive and advanced internet applications (websites) in a quick, non-stressful way. As the tagline itself states \*Ruby on Rails\* provides a balance between productivity and pleasure which comes from programming and enables programmer to create a beautiful code preferring convention over configuration.

Rails allows us to create internet applications based on databases using MVC (Model-View-Controller) pattern. It gives possibility to effectively use Ajax available in Views, then effectively use queries and answers in controllers (Controller) and build advanced business logic in models (Model).

Three principles always emphasised by the Rails environment are:

- \*DRY\* (Don't Repeat Yourself) - application code should be created in such way that one can use it many times; thanks to that one can avoid writing the same logic in different parts of a program

- \*Convention Over Configuration\* - the authors of the environment use many practical conventions and assumptions about applications, so that the programmer can avoid unnecessary and mundane (repetitive) configuration of his application and environment.

- \*Quick Implementation of Changes\* - integrated web server and interpreted language allow us to quickly observe changes made in code (no compilations and no need for restarts) which reduces the amount of time needed for developing applications. Emphasis on testing the code and using TDD techniques allow us to quickly update the application without the risk of errors or regressions.

### Duration

3 days

### Agenda

1. Introduction, basic applications, models
  - Introduction and history of Rails (in a nutshell)
  - Architecture of Model-View-Controller
  - Project elements in Rails environment
  - Project directory structure
  - Naming conventions
  - The idea of resource and REST conventions
  - ActiveRecord and ORM (Object-Relational Mapping)
  - Migrations
  - Model associations (has\_one, belongs\_to, etc)
  - Model validations
  - Callbacks in models
  - Using console while programming (Hirb-like extensions)
2. View and controller layer, templates
  - ActionController – controllers
  - Cookies and managing a session
  - Filters in controllers
  - Routes

### Ask for details

Phone +44 203 608 6289

info@alx.training

- ActionView – views
- Layouts and partials, helpers (decorators)
- Template languages (ERB, Haml, Slim)
- Short introduction to Twitter Bootstrap and style templates
- Ajax (JQuery and RJS templates)
- Generators and rake
- Various application environments (production/development/test)
- 3. Advanced technologies, the most popular gems
  - CSS management and Javascript in application (asset pipeline)
  - Briefly about Less, Sass and CoffeeScript
  - The security of WWW application
  - Rails: best practices
  - Recommended gems for repetitive tasks:
    - Authentication (Devise)
    - Authorisation (Cancan, Rolify)
    - Background jobs (Sidekiq, Resque)
    - Images and other attachments (Paperclip, Carrierwave)
    - Reporting exceptions (Errbit)
    - Testing (Guard, RSpec, Minitest)
  - Useful gems for application development
    - BetterErrors
    - Pry
    - Debugger
    - Faker
  - Rails application cache (caching)

### Ask for details

Phone +44 203 608 6289

info@alx.training

## Target audience and prerequisites

We assume, that participant understands at least basic programming concepts (variables, loops, conditions, functions, objects). Familiarity with any programming language will help a lot. Prior Ruby knowledge is not required. Language itself will be introduced during the course.

It's also recommended to have a good general understanding of concepts regarding creating websites. Below you will find the summary of requirements/recommendations for the initial knowledge and skills of training participants. Note, that familiarity with all the listed elements is not required, but the more you know, the easier the course will be to understand.

- knowledge of programming concepts and constructs
- basic knowledge of general concepts about creating WWW applications
- knowledge of HTML and/or CSS
- general understanding of HTTP protocol and topics like redirections, SSL, differences between GET and POST
- basic knowledge of Javascript language (and basics of JQuery) is recommended
- basic knowlegde of SQL and MySQL or Postgres database is recommended

## Certificates

Course participants receive completion certificates signed by ALX. Upon completing the course, participants will have the following skills/ knowledge:

- familiarity with Ruby programming language
- ability to build application with Ruby on Rails from the beginning
- managing RoR application's database through migrations

- ability to create advanced application logic through ActiveRecord relations
- basics of building user interface based on Ajax
- searching and using external libraries (Gems) in Rails applications
- basic understanding of TDD techniques idea
- ability of routes configuration in Rails applications
- understanding some essential, good practices connected with creating websites with Rails

### **Ask for details**

Phone +44 203 608 6289  
info@alx.training

### **Locations**

- Warsaw (English) – Jasna 14/16A
- Online (English) – your home, office or wherever you want
- any other location (London, UK, EU) on request

### **Price**

990 EUR

The price includes:

- course materials,
- snacks, coffee, tea and soft drinks,
- course completion certificate,
- one-time consultation with the instructor after course completion.