

**Czech University of Life Sciences in Prague**  
**Faculty of Economics and Management**  
**Department of Information Technologies**



**Diploma Thesis Abstract**

**Web Applications Development in Ruby on Rails**

**Author: Jiří Procházka**

**Supervisor: Ing. Pavel Šimek, PhD.**

© 2015 CULS in Prague

## Summary

The thesis deals with development of web applications with relational and document-oriented databases using Ruby programming language and Ruby on Rails framework. The theoretical part summarises information about Ruby programming language and Ruby on Rails. The next part introduces relational (PostgreSQL) and document-oriented (MongoDB) databases with their basic architecture and connection with Ruby on Rails. The last section from the literature review describes a few tools and utilities typically used for web applications development such as Bootstrap or Git.

The practical part is focused on developing a web application with PostgreSQL and MongoDB databases. It describes setup of the developer's machine environment as a preparation for the development. The development itself consists data model implementation with Ruby on Rails' generators and PostgreSQL. Afterwards, the relevant controllers and views are implemented in order to create the application logic. The design is created with Bootstrap framework. The following section describes the refactoring process, where PostgreSQL database was changed to MongoDB database. The application is covered by tests for both databases to test all of its functionalities. Lastly, the practical part deals with deployment to free cloud platforms (Heroku and OpenShift) and a description of the whole process.

**Keywords:** Ruby, Ruby on Rails, MongoDB, Mongoid, Bootstrap, PostgreSQL, OpenShift, Heroku, Web application, NoSQL, Git, SQL, RVM

## **Thesis objectives and methodology**

The thesis deals with a development of web applications with databases using Ruby programming language and the framework Ruby on Rails. The main purpose is to analyse problems connected with the process of the development and introduce few approaches to implement web applications. Partial goals of this thesis will include comparison of the relational and document oriented database and description of various tools and utilities commonly used for the development. The practical part of the thesis will cover real development, implementation and deployment of a web application using the theory from the literature review.

The methodology of the thesis is based on study and analysis of information resources. At first, the theoretical part summarised information about Ruby programming language and its features, followed by possibilities of extending Ruby with additional libraries. The next part dealt with the introduction of Ruby on Rails framework for web application development. Consequently, relational (PostgreSQL) and document-oriented (MongoDB) databases were introduced with basic architecture and connection with Ruby on Rails. The last part from the literature review described a few tools and utilities typically used for web applications development such are Bootstrap or Git.

The practical part is focused on developing a web application with database. At first, an assignment of the web application was formulated and formalised by diagrams and description. Afterwards, as a preparation for the development, it was necessary to set the developer's machine environment. The development itself began with data model implementation with Ruby on Rails' generators and PostgreSQL. Once the models were created the relevant controllers and views were implemented in order to match the application logic. The design was created with the aid of Bootstrap framework and author's experiences and skills. The last part of the development was the creation of tests covering the application.

The following section of the solution was to describe the refactoring process, where PostgreSQL was changed to MongoDB with corresponding modifications to models. The application was again covered by tests with the new database. Lastly, the practical part dealt with deployment to free cloud platforms and description of the whole process. The application in version with PostgreSQL was deployed to Heroku, and with MongoDB to OpenShift.

## Conclusion

Ruby on Rails confirmed its status as very productive and fast prototyping framework. Work with it was efficient and readable, with minimum configuration needed. Ruby itself has very simple syntax, which emphasises the ease of using Ruby on Rails. Since Ruby is interpreted language, Rails can offer a console interface, which helps the development in a way of testing methods and manipulation with data. The MVC pattern, which Rails framework incorporates, is the most popular for its structure and independence between individual parts. This was confirmed in the practical part, when all of the models were switched from PostgreSQL database to MongoDB. This transition did not require any additional changes to views or controllers, thus proving the independence and high abstraction of database layer. The framework also supports running several databases at the same time, which gives a space for various migrations or using advantages of different databases' concepts.

The application was developed with PostgreSQL at first and then refactored with MongoDB, thus using different databases within the same application. This enabled to compare both databases in the terms of implementation difficulty and performance. These databases are incorporating different concepts and technologies. It is difficult to compare it, since it differs a lot in certain areas. However, this thesis allowed observing particular advantages and disadvantages of both. PostgreSQL is considered as rather adult and mature database, with transactions support, which is very useful for financial records. It is sometimes too strict when it comes to saving data with various encoding and formats. PostgreSQL is more than 30 years older than MongoDB, therefore it provides wider support, comprehensive documentation and countless number of solved issues. However, in comparison to MongoDB it has several downsides. MongoDB has built-in scaling and sharding options that work out of the box, which PostgreSQL lacks.

If the database designer follows rules of normalization, relational databases tend to have queries with many joins, which rapidly slow down the execution time when the database contains numerous records. MongoDB provides document embedding, which is basically storing documents within one parent, thus receives all desired data from a single object with one query.

Fixed schema of relational databases versus dynamic fields of MongoDB is a controversial topic. It is sometimes desired and considered as an advantage, yet sometimes it can be a security and consistency risk. From the point of the development in the practical part of this thesis (and development in general), dynamic fields instead of fixed schema are definitely

an advantage, which speeds up the process. However, the production point of view might be the opposite.

Regarding the performance comparison in the application, there were no observable differences between PostgreSQL and MongoDB. The application is too small and simple, running in one server environment for such a comparison. Practical experiences showed that even within larger applications it is sometimes impossible to evaluate, which database is better. Both databases have some advantages and disadvantages, where the balance in between is dependent on data model and application requirements.

Lastly, the practical part included deployment to two different cloud platforms - Heroku and OpenShift. Both platforms were fairly easy to use and deploy. They provide similar functionality within the free plans. There were some minor differences of their services, but generally insignificant. Nevertheless, each platform has its main disadvantage that was noticed during the deployment. Heroku doesn't provide access to its filesystem, thus denying storage of any binary files there. As a substitution, Heroku offers linkage to other cloud services (such is Dropbox, Amazon S3, etc.) for the storage. OpenShift, on the other hand, offers around 1 GB of file system space. However, the service, its web pages and the whole administration was very slow and often terminating due to some errors. Nevertheless, considering that both of these cloud platforms are free of charge, the quality is more than sufficient.

The main output of this thesis is the custom tailored web application for the company from building industry, which can suit as a tradable product in the future. It also gives a described example of development process that can suit as manual for creating other web applications. Using SQL and NoSQL databases showed the flexibility of Ruby on Rails and the incorporated the MVC pattern. The thesis also theoretically and practically introduced various tools and utilities for web application development. The deployment and usage of two free cloud platforms gave an example how to publicly display and operate a web application without any costs.

The future of web applications will probably increase on its popularity since the tools and ways of development are becoming more productive, efficient and easy to use. An obvious benefit of web application is mainly the platform independence, which have enlarged its usage to mobile devices, tablets, etc. Technological progress enables to reduce costs on computer equipment, as well as server hosting or cloud platform rental. Thanks to that, the web application development has a very promising future.

## List of selected bibliography

BANKER, K. *MongoDB in Action*. 1st Edition. Shelter Island: Manning, 2012. 287 s. ISBN: 9781935182870.

CHODOROW, K. *MongoDB: The Definitive Guide*. 2nd Edition. Sebastopol: O'Reilly Media, 2013. 409 s. ISBN: 978-1-449-34468-9.

FLANAGAN, D. a Y. MATSUMOTO. *The Ruby Programming Language*. 1st edition. Sebastopol: O'Reilly Media, Inc., 2008. 429 s. ISBN 978-0-596-51617-8.

GAMBLE, A., C. CARNEIRO a R. A. BARAZI. *Beginning Rails 4*. 3rd Edition. Berkley: Apress, 2013. 303 s. ISBN 978-1-4302-6034-9.

LOELIGER, J. a M. MCCULLOUGH. *Version Control with Git*. 2nd Edition. Sebastopol: O'Reilly Media, Inc., 2012. 434 s. ISBN: 978-1-449-31638-9.

MATTHEW, N. a R. STONES. *Beginning Databases with PostgreSQL*. 2nd Edition. Berkley: Apress, 2005. 637 s.

REGE, G. *Ruby and MongoDB Web Development Beginner's Guide*. 1st Edition. Birmingham: Packt Pub., 2012. 310 s. ISBN: 978-1-84951-502-3.

RUBY, S., D. THOMAS a D. H. HANSSON. *Agile Web Development with Rails 4*. 1st Edition. Raleigh: Pragmatic Bookshelf, 2013. 434 s. ISBN 978-1-937785-56-7.

SPURLOCK, J. *Bootstrap*. 1st Edition. Sebastopol: O'Reilly Media, 2013. 109 s. ISBN: 978-1-449-34391-0.

THOMAS, D., C. FOWLER a A. HUNT. *Programming Ruby 1.9 & 2.0*. 3rd Edition. Raleigh: The Pragmatic Bookshelf, 2013. 863 s. ISBN 978-1-93778-549-9.