Ruby on Rails Framework On Agile Developement

Dendy Suprihady/ 13514070
Informatics Engineering Program
School of Electrical Engineering and Informatics
Bandung Institute of Technology, Jl. Ganesha 10 Bandung 40132, Indonesia
dendyliu666@yahoo.com

***Abstract*— This paper is an introduction to Ruby and to the Ruby on Rails framework. Ruby is the name of the programming language, while Ruby on Rails is the name of a framework for developing data-based websites using Ruby. The framework is frequently referred to as simply Rails, which is how I most often refer to it in this paper. In print, although not when spoken, the framework is also sometimes referred to as RoR.**

Keywords*— *Ruby on Rails (RoR), Rails Framework, Agile development, MVC architecture.

I. INTRODUCTION

In the recent years there has been a rising of a new style of software framework being referred to as agile-rails framework . Since 1990s it developed out of the efforts of many peoples who dealt with software process , discovered them having, furthermore anticipated for a new framework to web development. Ruby on Rails is an open source web based application framework composed using ruby programming language. It is focused on Model-View-Controller architecture. Rails framework is sufficiently basic to require just a couple of lines of code to raise a whole web application with less configuration

II. BASE THEORIES

A. *Ruby (The Programming Language)*

Ruby is a object-oriented programming language that is translated. Ruby started during the mid-1990s in Japan and was initially developed and designed by Yukihiro. It has syntax similar to Perl and semantics to Smalltalk. Ruby itself was developed in C dialect. Ruby was made with less developer work and also possible confusion. In light of these reasons, the Rails framework was written in ruby. Ruby has started to become popular worldwide in the recent years.

B. *Rails(The Framework)*

Rails are a web framework built on Ruby, consequently the name Ruby on Rails. It empowers the programmer to easily create progressed database-driven websites using scaffolding and code generation, taking convention over configuration, which means that if you adhere to a certain set of conventions, a lot of features will work right out of the box with almost little code. Rails are an open source Ruby framework for creating web applications.

C. *Ruby on Rails (RoR)*

Ruby on Rails is a web framework written in Ruby. Ruby on Rails makes it simple to assemble a database-backend sustains web application that uses the language Ruby

III. ANALYSIS

A. *Agile Design Architecture*

Agile design architecture involves gathering of software development methodologies based on iterative and incremental development, wherein requirements and solutions go ahead through collaboration between a self and cross-functional team. Agile methods break major tasks into smaller number of subtasks called increments with minimal planning, and don't specifically involve long-term planning. Iterations are shorter timeline that ordinarily takes from one to four weeks at the maximum. Each iteration make use of team working through a full development cycle which incorporates: planning, requirements analysis, design, coding, unit testing, and acceptance testing when a working product is at last shown to stakeholders. This helps to minimize overall risk, and lets the project adjust to changes quickly. Iteration may not add enough functionality in order to warrant a business sector release, but the objective is to have an accessible release with as much as minimum bugs at the end of every iterations. Multiple iterations may be required to release a product or features. The below figure demonstrates the steps in Agile design architecture which concentrate on iteration and adaptable to change.

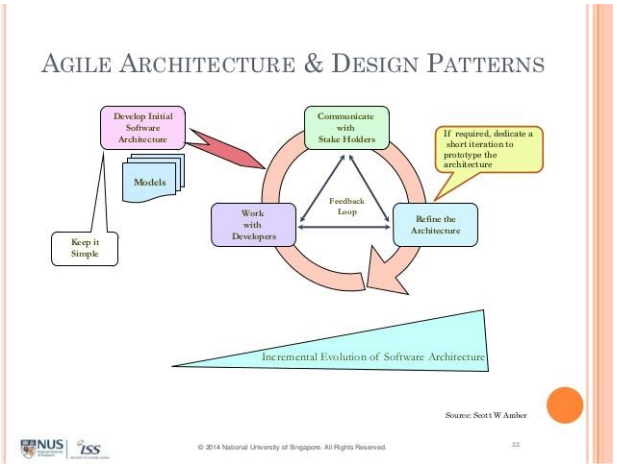


Figure 1. Agile Design Architecture

B. Rails Design Architecture

Rails are focused on a programming pattern called MVC, which remains for Model-View-Controller. The Goal of Rail design outline is to discrete organization of data (model) from UI & presentation (view) by introducing controller. One of the important feature of Ruby on Rails is it depend on Model-View-Controller architecture (MVC). The main advantage of MVC is to have the detachment of Business logic from client-user interface. Other advantage incorporates ease of keeping code, DRY and making it clear where distinctive sorts of code belong for less demanding upkeep.

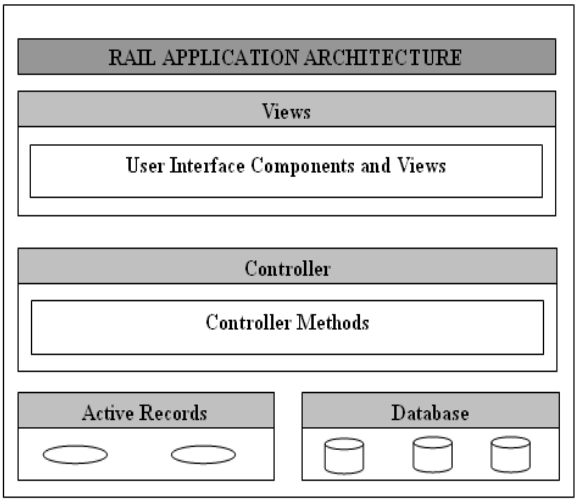


Figure 2. MVC Architecture

- **Model**: Model stores the information (data) about each table of database. If there should arise an occurrence of Rails, do some primary validation of data before putting into database and one table relate to one model.
- **View**: View, for the most part the interface of the application. If there should be an occurrence of Rails, Views are often HTML pages with some inserted ruby code. Views handle the job of giving the data to web browser.
- **Controller**: Controllers gives the "glue" in the middle models and views. In Rails, Controller fundamentally handles the incoming request from the web browser interrogating the models for data and passing that data on to the views for presentation.

C. The Component Of Rails

Architecture of Rails framework is illustrated in Figure 3. Rails architecture comprises of two noteworthy elements which incorporates Active Record and Action Pack. Action Record is an Object Relational Mapping (ORM) layer which handles the Model element of the MVC application. This layer covers the data table rows as model objects to the rest of the application. Action Pack handles the view and controller elements of the MVC application. Action Pack comprises of two parts viz. Action View and Action Controller which handle View and Controller of MVC application individually. Each of the architectural elements in the picture above is detailed below:

- User component – Rails web application is accessed by browsers using different protocol interfaces such as HTTP, FTP or expended as a web services using SOAP.
- Web Server – The developed Rails application dwells on a Web Server, which handles request from the client-user component and advances them to the Dispatcher.
- Dispatcher – Dispatcher invokes proper controller focused on the current user request.
- Controller –Controller processes the user request and invokes proper action. The action works together with appropriate model objects and prepares the response.

In the event that the request was from the browser, the response will be rendered through Action View Component else a suitable response is relayed by either the Action Web services or the Action Mailer Component.

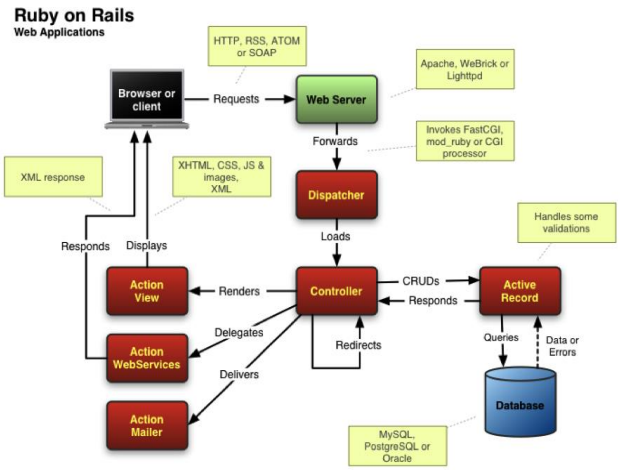


Figure 3. Architecture Model for Ruby on Rails (RoR) Web Application Implementation.

IV. HOW DOES RUBY ON RAILS ENABLE AGILE DEVELOPMENT

What makes rails agility which thus infers how Ruby on Rails empowers agile development? In this section might elaborate the characteristics of Rails framework that make it most appropriate for agile methodologies. MVC architecture - which permits web professionals to partition business logic, user presentation and control features and encourage the process of web development.

CRUD (Create, Read, Update, Delete) - In the heart of the Rails routing engine lies CRUD, or Create, Read, Update,

Delete, which originates from a notion that all web pages are comprised of these four actions. For example, can make a product, view, or read a client-user, edit, or update a category, and delete an order.

If follow this pattern, will accomplish a great deal of work done. For example, all it takes to make all of these actions for a product is to make the routing engine that have an entity called product, and it will wire up all the routing logic that ties the request to the matching controller.

DRY – "Don't Repeat Yourself" – suggests that writing the same code over and over again is a terrible thing.

Convention Over Configuration – implies that Rails makes presumptions about what need to do and how are going to do it, rather than letting to change every little thing through interminable configuration files.

REST (pattern) - is the best pattern for web applications for sorting the application around resources and standard HTTP is the fastest approach to go.

Scaffolding - often make transitory code in the early phases of development to help get an application up rapidly and perceive how major components work together. Rails automatically make a great part of the needed scaffolding.

Portable programming skills – Team managers will admire the way that Rails minimizes configuration and encourages standardization. This empowers the programming skills more portable.

Rapid feedback loop – In Rails yet get almost-instant feedback as with code. Thus consequences of the progress made can be immediately seen improving the overall client-customer experience empowering changes made instantly as well.

A great number of software development companies strive to make the procedure of web application development less tedious and more flexible; hence, such organizations and its web specialists take advantage of Ruby on Rails technology, which allows them to utilize the principles of agile development.

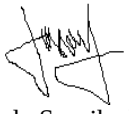
V. CONCLUSION AND FUTURE ENHANCEMENT

Ruby on Rails as an open source web application development framework focused around Ruby programming language and used extensively by agile programmer teams has been pretty popular for rapid web application development. Two interesting aspects of RoR stand out "the convention over configuration" wherein programmers specify the unusual configuration as well as "Don't Repeat Yourself" wherein information used is unique and not duplicated.

Rails are a perfect platform for Agile development practices. Keeping in view its agreement with the Agile Manifesto, it is not difficult to foresee that Rails will be the future of Agile development. As a fairly new web framework, Ruby on Rails has not yet seen the prominence of a number of today's current standards such as Java or .NET based frameworks. This does not imply that Ruby on Rails is unequipped of handling real world applications, as it can and has been taking care of wide range of dynamic web-support projects. Ruby on Rails is a extraordinary framework for creating progressed web applications writing very little code in comparison. Recommend it for almost any kind of application, and particularly for prototyping. To have a firm conclusion that a blog can be created within a moment using Ruby on Rails. Due to its straightforward nature and simple to use features as well as the speed of project completion.

ACKNOWLEDGMENT

I sincerely thankful to my lecture Mr. Rinaldi Munir , Who help me a lot in making of this assignment. I have learned a lot of different techniques in writing the paper about my long-life learning.



Dendy Suprihady/13514070

REFERENCES

[1] Nerur S, Mahapatra R & Mangalraj G (2005), Challenges of migrating to agile methodologies Communication of the ACM, Vol. 48 No. 5, Pages 72-78 New York USA.

[2] Martin Fowler "New Methodology" , http://martinfowler .com/ articles/ newMethodology.html Vol. 1 de 17 (19/08/2010).

[3] Andrew Mcdonald, "Agile Web Engineering (AWE) Process: Perceptions Within A Fortune 500 Financial Services Company" Journal Of Web Engineering, Vol. 4, No.4 (2005) 283-312.

[4] http://en.wikipedia.org/wiki/Agile_software_development..

[5] Osmar Zaine, "Report on Ruby on Rails", CMPUT 410, November 16, 2007.

[6] Osmar Zaine, "Report on Ruby on Rails", CMPUT 410, November 16, 2007.

[7] Qumer., & Sellers, An evaluation of the degree of agility in six agile methods and its implacability for method engineering. Information and Software Technology, (2008) 280-295.

[8] Phalnikar, Deshpand, &Joshi, Applying Agile Principles for Distributed Software Development - International Conference on Advances in Computing and Communications01/2009; DOI:10.1109/ICACC.2009.93 ISBN: 978-0-7695-3516-6.

[9] Judy, Simon & Schuster, Agile principles and ethical conduct. Proceedings of the 42nd Hawaii International Conference on System Sciences-2009.

[10] Malik Hneif, Siew Hock Ow, "Review Of Agile Methodologies In Software Development", International Journal of Research and Reviews in Applied Sciences ISSN: 2076-734X, EISSN: 2076-7366 Volume 1, Issue 1(October 2009).

[11] Asad Awan, "Agile Web Development with Rails – An Overview", Research article Friday 16. of January 2009.

[12] Sudipto Das, Vinod Kone "Ruby under Scanner: Comparison with Java" University of California, Santa Barbara 2010

[13] João M. Fernandes and Mauro Almeida, "A Technique to Classify and Compare Agile Methods" (Eds.): XP 2010, International Journal of Computer Applications (0975 – 8887) Volume 112 – No. 1, February 2015 11 LNBIP 48, pp. 385–386, 2010. Springer-Verlag Berlin Heidelberg.

[14] Dwayne Read, "Going Agile- A Case Study", 2010

[15] Aspire Systems, "Ruby on Rais-Building Dynamic Web Products Faster", 2011. Qumer., & Sellers, An evaluation of the degree of agility in six agile methods and its implacability for method engineering. Information and Software Technology, (2008) 280-295.