

Ruby Association

Conductor

Final Report

Laura Garcia Casadiego

February 28, 2014

This report summarizes the work I have done during the 2013 Ruby Grant period and specify the social impact and future work in mind.

These were the goals:

1. Add tests to Project.
2. Allow to run bundler using web UI.
3. Allow to run migrations and other DB tasks from UI.
4. Allow to run only tests of certain type (model tests, controller tests, integration...).
5. Add a code browser and editor.
6. Improve the UI/UX, the initial prototype was focused more in code side than the design.
7. Integrate the live development features of Hocus Pocus gem (scaffold generation in missing routes, link for live HTML editing...).

Add test to the project

The first version of conductor doesn't had test in the project. I added all test to the current implementation of Conductor. I deleted some tests that were added in the intermediate report because the models were removed.

Install Gems (with bundler)

A Gemfile controller was included that runs the command “bundle install” and shows the output in install Gems section. Also it can open the Gemfile file to modify the gems.

Run Migrations

I added a Migrations controller that shows the list of migrations of the database and allows to change the status to up or down and also shows the current version of migrations.

Database Task

Database view has buttons to run db:create, db:drop, db:seed and db:setup, also Migrations view has a button to run db:rollback.

Run test of certain type

In tests controller I added some actions that run tests of different types such as: migrations tests, controllers tests, models tests, integration tests and all tests.

Code browser and editor

The code editor let you browse the files and directories of the applications, open, edit, save changes and create new files. Have highlights for different types of languages. I used ace.io JavaScript library in Conductor for a an optimal data visualization.

Improve the web UI

Conductor was initiated by David Heinemeier Hansson several years ago and still maintained the same style. Changes such as font style and spacing used for the contents were made, but still use a style similar to the Rails default page information.

Integrate Hocus Pocus

Hocus Pocus is a gem created by Akira Matsuda, for more information https://github.com/amatsuda/hocus_pocus . It has four engines:

Editor provides the option to edit the HTML files in the browser . At the top right of each window there is a button to display the editor for the current view. This feature is now available in Conductor.

Recorder and Generator are also available in the gem now. Conductor already had a scaffold generator, to don't repeat the existing code, Conductor redirects to Conductor's Scaffold Generator when a route is missing like the functionality of Generator in Hocus Pocus. The Recorder works in the same way as in Hocus Pocus, at the top of each window beside the editor.

Command Line let you run a command for the console in the browser. Conductor has a link to Web-console, a gem that simulates a console and has a better performance that the Command Line engine. For more information about web-console go to <https://github.com/rails/web-console> .

In the sections as Gemfile Install and Run Test, I used websocket to improve the way the console output is displayed. Tubesock websocket works for concurrents servers such as Puma and Unicorn.

Social Impact

The goal of Conductor is to provide assistance to all Rails users during the development of their applications; this is specially useful for new users that aren't familiar with the command line utilities used typically in the terminal during the development (rails commands, rake tasks, bundler, etc.). Conductor also will teach to users how to do through the terminal everything that they do through the Web UI. To achieve this goal, a message with the console command for each feature was included.

A group of 3 students of computer science tested Conductor for developing a small project with Ruby on Rails, two students of this students had no experience previously with the framework. The Features that were considered most useful:

- Comfort using command frequently as bundle install and generators.
- The Hocus_Pocus Editor is useful for making small changes in views.

- One student considered better to use Conductor because he does not use the bash.
- It is very useful learn the commands using Conductor because for a newbie are many new commands to memorize.
- A code editor in browser that allows you to make small changes while testing your application
- Quick access to Ruby API and Ruby Guides for quick reference.

Besides, In Barranquilla have been organizing a Rails Girls Event, where I'm organizer, the main idea of the event is to teach to the attendees about the Framework and introduce the newer girls programmers to the web programming world. The organizers decided to include Conductor as a teaching tool in the event.

Future Work

Due that Conductor can be very useful for people who begin to learn Rails, in the future could be integrated more information oriented to help learning Ruby and Ruby on Rails.

Currently in rubygems.org there is a gem called Conductor but with a different functionality. Wherefore be published with the name rails-conductor to continue with the original name.

Site: <https://github.com/rails/conductor>