## Revised Level 3 Curriculum

## Day 1

* Morning 1 - Environment & Source Control
  + RVM
  + Bundler
  + Common Git Practices
  + Heroku Configuration / Setup
* Morning 2 - Routes
  + Request Cycle & Component Roles
  + REST Structure
  + Router:  
    Basic resources, singleton resources, root, named paths, custom actions, nested resources
* Afternoon 1 - Internal Testing
  + RSpec and BDD
  + RSpec practices: let, before, after, describe
  + Sample data with Fabrication
  + Coverage measurement with CoverMe
* Afternoon 2 - External Testing
  + Capybara with Rack::Test
  + Capybara with Selenium / HTMLUnit
  + Continuous Integration
  + Testing AJAX / JavaScript

## Day 2

* Morning 1 - Debugging
  + Manipulating the logs
  + Raising exceptions
  + Ruby 1.9’s debugger
  + Integrated debugging with RubyMine
  + Heroku Add-ons: Hoptoad and Exceptional
* Morning 2 - Models
  + Reviewing relationships
  + Validations
  + Polymorphism
  + Legacy/Non-conformant databases
  + Database locking
  + Transactions
  + Non-persistent Models
* Afternoon 1 - Business Logic
  + Processor Model Design Pattern
  + Pulling out modules to share code
  + ActiveSupport::Concern
* Afternoon 2 - Controllers
  + Handling parameters
  + Render and redirect
  + Managing sessions
  + Imitating conversations

## Day 3

* Morning 1 - Views
  + HAML versus ERB
  + Partial Techniques (collections, variables)
  + Pagination
  + Friendly URLs
  + Managing the Flash
* Morning 2 - JavaScript & AJAX
  + Changing over to jQuery
  + Rails built-in JavaScript helpers
  + Patterns for managing JavaScript code
* Afternoon 1 - Providing Web Services
  + Exposing APIs from the controller
  + Encoding data in the model
* Afternoon 2 - Consuming Web Services
  + Consuming REST with ActiveResource
  + Consuming SOAP with Savon
  + Keeping the source contained

## Day 4

* Morning 1 - Performance
  + Consolidating queries
  + Caching with Memcache & Redis
  + Creating workers with DelayedJob
  + View fragment caching
* Morning 2 - Systems Management
  + Scheduling with Cron
  + Managing credentials
  + Flexible application configuration
  + Monitoring performance
* Afternoon 1 - Authentication
  + Local authentication with Devise
  + Remote authentication with OmniAuth
* Afternoon 2 - Authorization
  + Implementing Declarative Authorization
  + Structuring roles
  + Controlling view, controller, and model access

## Day 5

* Small Topics & Cleanup
  + Finishing anything cut short from the week
  + Implementing Search
  + Configuring Heroku features

## Trimmed Topics

* Other persistence options
* Implementing presenter models
* Providing Web Services - Writing customized view templates
* Creating workers with resque
* Outside-in testing (black-box testing)