Beginning Ruby on Rails

Session Eight

Deployment

- Git
- Heroku
- Capistrano
- Security
- Performance

git

Written by Linux Torvalds - a despicable person

Getting Started

- Create an empty repository:
  - git init
- Add all files:
  - git add .
- Commit:
  - git commit -m “Initial commit.”

Simple Usage

- Make changes to files
- Add all changed files:
  - git add .
- Commit changes:
  - git commit -m “commit message”

Other Commands

- See changed and new files:
  - git status
- See detailed changes:
  - git diff <filename>
- Clone someone else's repository:
  - git clone <url>

Remotes

- Add a remote:
  - git remote add <name> <url>
- Push changes to the remote server:
  - git push <name> <refspec>
- Pull changes from the remote server:
  - git pull <name>

Heroku

One step deploy

What's a Heroku?

- A platform as a service (PAAS) for running "cloud" apps.
- An extensive collection of add-ons.
- Free to start, but can get expensive depending on resources needed.
- Recently hired Matz.

Deploying to Heroku

- Prerequisites:
  - Sign Up at http://www.heroku.com
  - Install the heroku gem
    - gem install heroku
  - Set up SSH keys

SSH Keys

- On Mac or Linux:
  - ssh-keygen -t rsa -C “email”
- RailsInstaller creates keys automatically
- Or, you can open “Git Bash” and use the same command as Linux or Mac

Deploying to Heroku

- Create Heroku application
  - cd blog
  - heroku create
- git push heroku master
- heroku rake db:migrate

Capistrano

Deploy anywhere

Getting Started

- Install the Capistrano gem:
  - gem install capistrano
- “Capify” your application:
  - capify .

Configuration

- Edit config/deploy.rb
- Details are environment and app specific:
  - ‘set’ variables
  - ‘role’ lists servers
  - ‘namespace’ and ‘task’ define work to perform on the server.

Setup

- Get the server ready for your app:
  - cap deploy:setup
- Make sure everything is set up correctly:
  - cap deploy:check

Deploy

- Deploy the latest code and restart the app:
  - cap deploy
- Deploy, run pending database migrations and then restart:
  - cap deploy:migrations

Security

Beyond Authentication and Authorization

Mass Assignment

- Use attr_accessible
- List database fields that should be available for modification by users
- All other fields will be protected from assignment by ‘new’ and ‘update_attributes’

Privilege Escalation

- Humans are naturally curious
- Some users will change URLs in an attempt to see other users' data
- Always use proper authorization

Performance

A few quick tips for faster apps

N+1 Queries

- Index creates 7 queries for 5 issues
- Use eager loading
  - Issue.includes(:notes)
- Use “length” instead of “count” in the view
  - count always makes a query

Caching

- Rails offers 3 types "out of the box"
  - page, action, and fragment caching
  - caches_page :index
- Understand expiration
  - expire_page :action => :index

Homework

We're done, you're free

Feedback

- I would love to hear your thoughts:
  - What was confusing?
  - Did I go too fast / too slow?
  - Did I leave anything out that you really wanted to see covered?

Keep Learning

- Books:
  - Agile Web Development with Rails
  - Programming Ruby (The Pickaxe)
  - The Rails 3 Way
  - Rails 3 In Action

Keep Learning

- Web Sites:
  - http://railscasts.com
  - http://guides.rubyonrails.org
  - http://www.codeschool.com
  - http://ruby.railstutorial.org

Keep In Touch

- I love talking about Rails in my spare time:
  - Send me an e-mail or tweet
  - Follow my blog
- Connect with me on LinkedIn:
  - Recommend me?