Case Study - Grails in the Real World™

By Greg L. Turnquist

Friday, September 13, 13

Who am I?

- Test-bitten script junky

- Always looking for the right tool for the job

- Founded Nashville JUG in 2010 (almost 300 members)

- Created Spring Python in 2006

- Author of **Spring Python 1.1** and **Python Testing Cookbook** and reviewed several others

- Twitter: @gregturn

- Blog: GregLTurnquist.com

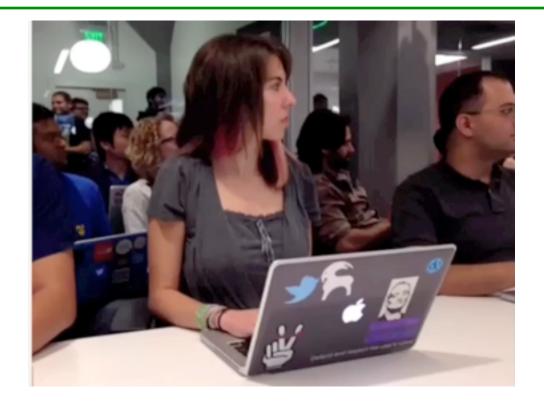# Case Study - Grails in the Real World™

Questions?
Keep 'em coming!

Engage!

Friday, September 13, 13

Case Study - Grails in the Real World™

- Register by email
- Plug in Spring Security
- Code specialized REST endpoints
- Cache files with MongoDB's GridFS
- Polish UI with Bootstrap + jQuery
- Protect users from themselves

springone 2gx

Maven proxy server

Your project

Friday, September 13, 13

Register by email

Register by email

Pros | Cons

Pros
- Easy to confirm
- Easy to update
- Easy to manage
- Built in, out-of-band confirmation

Cons
- Publicly visible emails susceptible to spam
- Corporate accounts recyclable
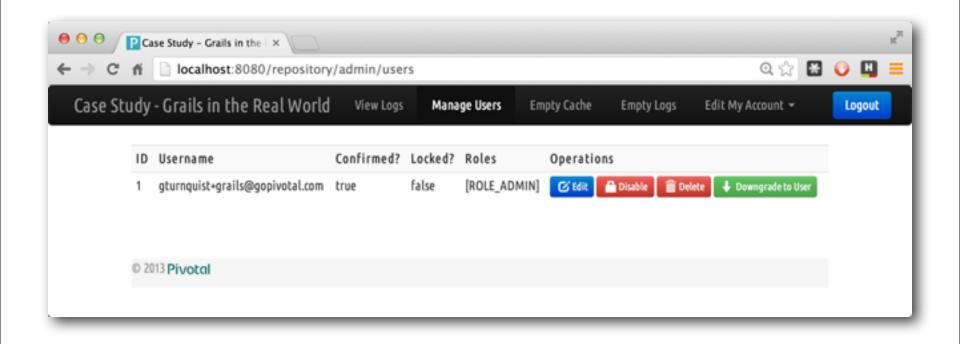
Plug in Spring Security

- Easy to get started
- Quickly tag backend services/controllers
- Easily protect front end HTML with <sec> tags
- Well supported by community and books
- Corporate backing

Plug in Spring Security

Code specialized REST endpoints

- Easy to create custom endpoints
  - Edit UrlMappings.groovy
- Pick your strategy
  - Hashed values or authenticated endpoint?
- BTW: Grails 2.3 may render some of this obsolete
  - Comes with new built-in conventions
  - But...if you do something different, can always add it on!

Cache files in MongoDB's GridFS

Cache files in MongoDB's GridFS

- Stash downloaded files in GridFS

- Speed up user experience

- Groovy makes GridFS API a snap to use

- Implement FIFO - First In First Out

Polish UI with Bootstrap + jQuery

Junior developer:
"I can build that!"

Senior developer:
"Build that, someone else has."

Polish UI with Bootstrap + jQuery

- Bootstrap + jQuery make me look good!
- Use HTML5 form validation features
- Find themes. It's okay to pay for good ones!
- Hire a top quality UI person

Protect users from themselves

Avoid bad data!

- Double up on customer data inputs
- Consider buying "good data"
  - Nationwide address list, zip codes, CLLI/LID/FAC database
  - Steer users towards this data with prompts
- Assume the worst
- Having a demented team member can help!

Protect users from themselves

- Confirm registration by requiring re-entry of password
- Send address updates to old & new addresses
- Validate, validate, and validate!

Questions?

Learn More. Stay Connected.

Thanks!

Twitter: twitter.com/gregturn

Code: github.com/gregturn/grails-real-world