Rails from the Ground Up!

with Jared Richardson
RoleModel Software

A Misc Deck

A place to drop in miscellaneous bits

Easier to keep track of

Your Turn

Put a web wrapper on the War server

Allow a user to play the server

Timed game

Your Turn

- Put an HTML front end on the War server
- Let 2 users play
- Timed games
- Track win/loss record by IP address
- Need auto-refresh or AJAX to update each player when it’s their turn

Tic Tac Toe

Write Tic Tac Toe

Create a board object

First code a console version

Later in the class, do a web front-end

Data storage hints?

```
[[a,b], [c,d], [e,f]]
```

```
{ 1 => :e, 2 => :x, .... }
```

Outline/Overview

Create an overview slide for day one

Maybe an overview per day

Devise and CanCan

Roles, log in, etc

.rvmrc

start to make the sample war project more robusrt

add in a .rvmrc file

update Gemfile (as needed)

Ruby

CONSTANTS

Accessing CONSTANTS Platform::CR, etc

Custom Rake Tasks

Review the writing of Rake tasks

Be sure it's covered earlier

Add rake coding samples

Branching

branching & merging with git

generate scaffold

```
rails generate scaffold FastestGame
player:string score:integer
```

A scaffold in Rails is a full set of model, database migration for that model, controller to manipulate it, views to view and manipulate the data, and a test suite for each of the above.

http://guides.rubyonrails.org/command_line.html#rails-generate

rails console

More Active Record

basic intro in Rails_1

Have another slide deck

AR errors http://guides.rubyonrails.org/active_record_validations_callbacks.html#working-with-validation-errors

Browser Tools

YSlow

FireBug

Tamperdata

PageSpeed (developers.google.com/pagespeed

unicorn

```
gem install unicorn

unicorn_rails
```

Rendering/ERB

simple HTML pages

HAML

Coffee Script

Blocks and Closures

http://www.artima.com/intv/closures2.html

Re-create Ruby in the Browser

Add 2 windows/divs/etc

Submit button

Read the result as it's posted

Eval it

Print result in second frame

modules

yield