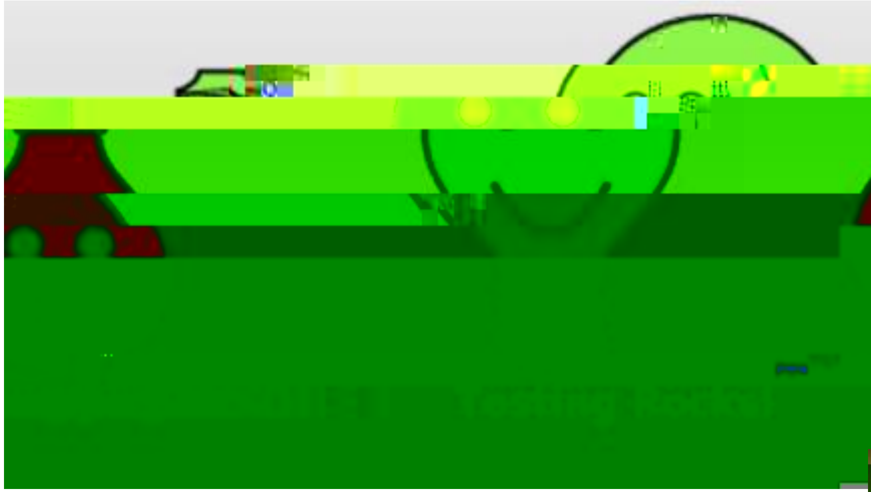
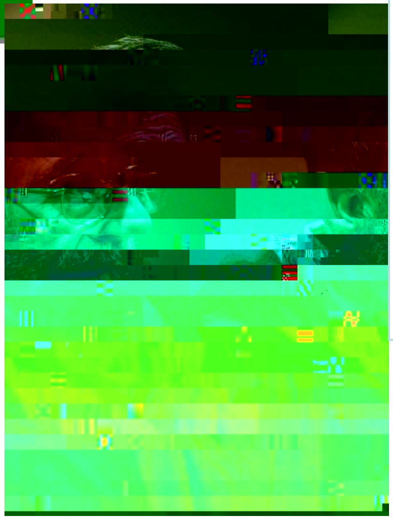
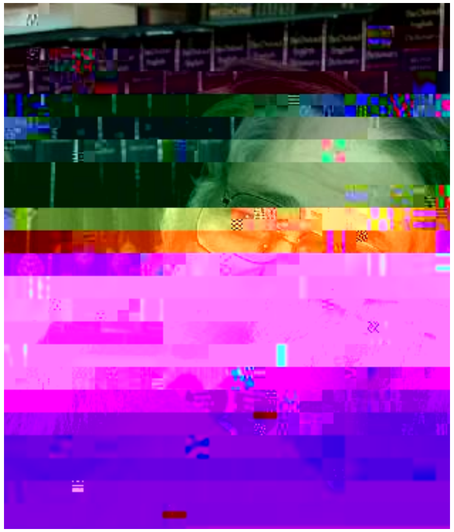




Testing can never demonstrate the ______ of errors in software, only their _______



BDD+TDD: The Big Picture

Cucumber & RSpec

Å

Unit tests should be FIRST

ÅFast: run (subset of) tests quicklmick′ l¾î ᵀ

Example: calling TMDb

Å

TDD for the Controller action: Setup

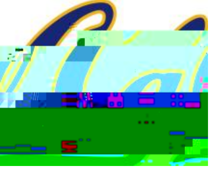
Å

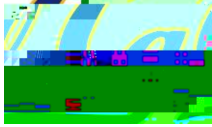
Seams

ÅA place where you can change your appôs *behavior* without editing the *code*.(Michael Feathers, *Working Effectively With Legacy Code*)

ÅUseful for testing:

How to make this spec green?





Where we are & where weôre going: ñoutside inò development

ÅFocus: write *expectations* that drive development of controller method

ÅDiscovered: must *collaborate* w/model method

ÅUse outside-in recursively: *stub* model method in this test, write it later

ÅKey idea: *break dependency* between method under test & its collaborat7rs

ÅKey concept:

Checking for rendering

- After post :search_tmdb, response() method returns } A] m rolles *response object*
- render_template matcher can check what view the controller tried to render

http://pastebin.com/C2x13z8M

- Note that this view has to exist!
- post :search_tmdb will try to do the whole MVC flow, including rendering the view
- hence, controller specs can be viewed as *functional testing*

Two new seam concepts

Åstub

Åsimilar to should_receive, but not expectation

Åand_return optionally controls return value

Åmock: create dumb ñstunt doubleòobject

Å

A mock and an expectation; an expectation. A seam and a