

1AAM

**GROOT**

**Okay, I'm not**

# Scott Smith



MEDICARE

HEALTH INSURANCE

1-800-MEDICARE (1-800-633-4227)

- Old Fart Developer

NAME OF BENEFICIARY

- Blog: [blog.scottnelsonsmith.com](http://blog.scottnelsonsmith.com)

JOHN DOE

MEDICARE CLAIM NUMBER

- Twitter: [\\_ofd](https://twitter.com/_ofd)

000-00-00000-A

SEX

- Email: [scottnelsonsmith@gmail.com](mailto:scottnelsonsmith@gmail.com)

MALE

IS ENTITLED TO

EFFECTIVE DATE

- GitHub: [oldfartdeveloper](https://github.com/oldfartdeveloper)

HOSPITAL (PART A)

01-01-2007

MEDICAL (PART B)

01-01-2007

SIGN



**SPOILED RAILS DEVELOPER**



**which is why I was  
attracted to Ember-  
CLI**



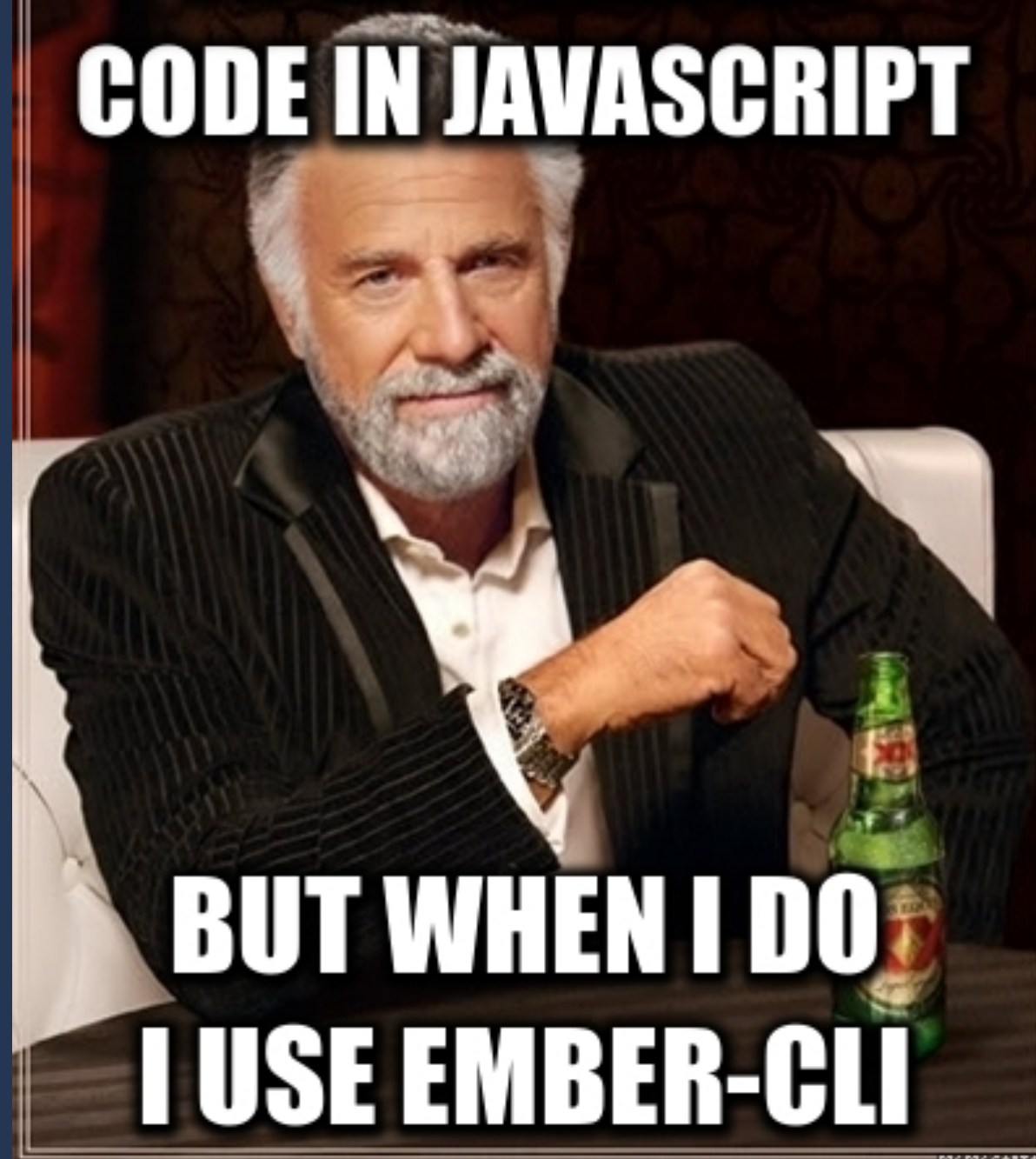
# Ember-CLI

- Ember.js command line utility
  - Broccoli-based pipeline
  - ES6 module syntax to facilitate many source files
  - Test file management
  - Dependency management

**this is how software  
is developed in  
the real world!**



**I DON'T ALWAYS  
CODE IN JAVASCRIPT**



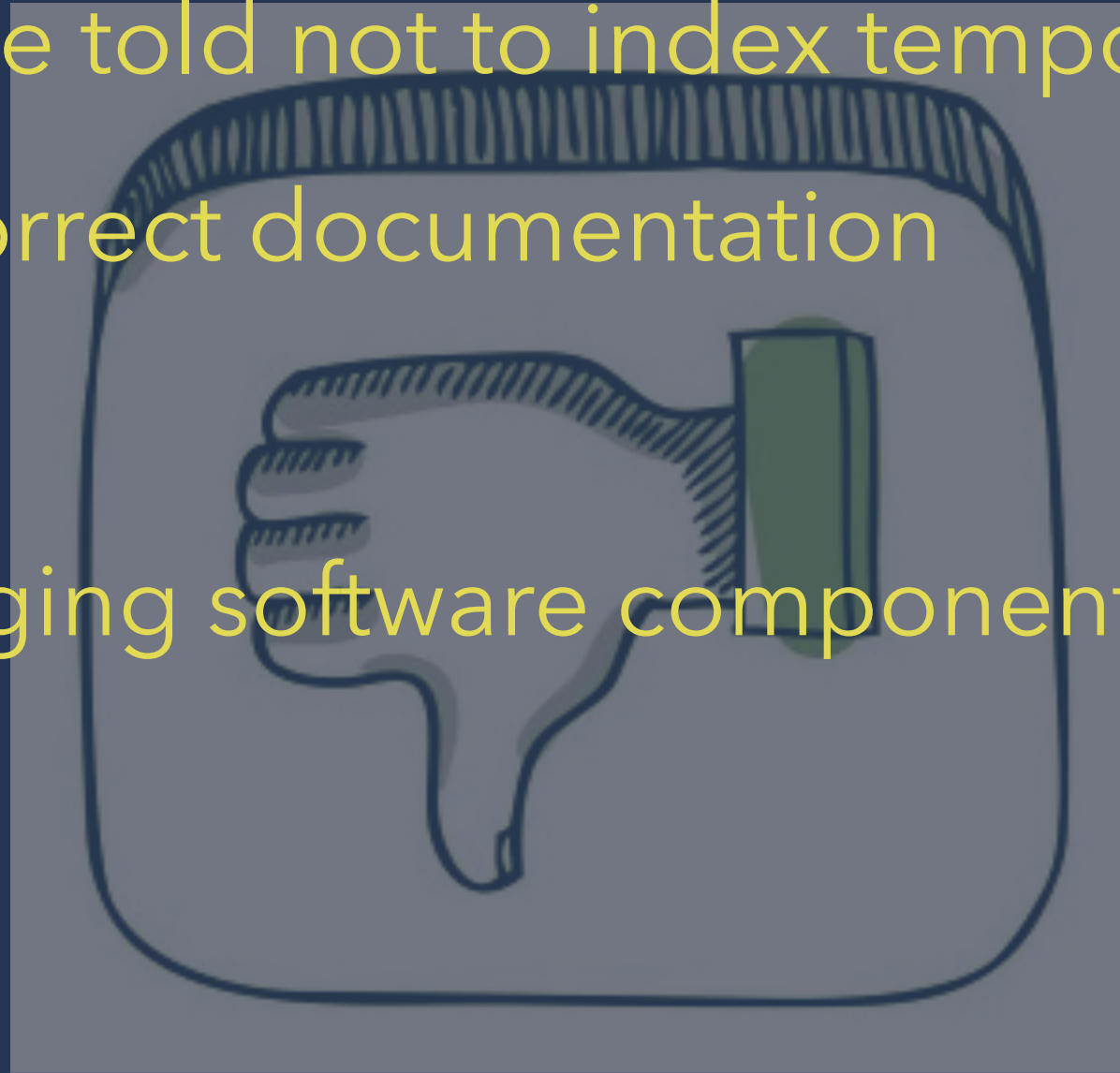
**BUT WHEN I DO  
I USE EMBER-CLI**

MEMECAPTAIN

- Really convenient once set up
- Efficient Broccoli asset refresh after every file save.
- Automatic test execution, including multiple browsers on Testem



- Pain to set up
  - IntelliJ must be told not to index temporary files
- Incomplete/incorrect documentation
- "stable" alpha
  - Rapidly changing software components/changing APIs



# Demo

**So run the damn app!**

# Fixture for Integration Testing



# Speakers

```
export default function initialSpeakers() {  
  return [  
    { id: 1, name: 'Bugs Bunny', presentation_ids: [1, 2] },  
    { id: 2, name: 'Wile E. Coyote', presentation_ids: [3] },  
    { id: 3, name: 'Yosemite Sam', presentation_ids: [4, 5, 6] }  
  ];  
}
```

# Presentations

```
export default function initialPresentations() {  
  return [  
    { id: 1, title: "What's up with Docs?", speaker_id: 1 },  
    { id: 2, title: "Of course, you know, this means war.", speaker_id: 1 },  
    { id: 3, title: "Getting the most from the Acme catalog.", speaker_id: 2 },  
    { id: 4, title: "Shaaaad up!", speaker_id: 3 },  
    { id: 5, title: "Ah hates rabbits.", speaker_id: 3 },  
    { id: 6, title: "The Great horni-todes", speaker_id: 3 }  
  ];  
}
```

# Server Simulation

```
export default function fixtureServer(speakers, presentations) {
  return new Pretender(function () {
    this.get('/api/speakers', function (request) {
      return [200, {"Content-Type": "application/json"}, JSON.stringify({speakers: speakers, presentations: presentations})];
    });

    this.get('/api/speakers/:id', function (request) {
      // Do stuff
      return [200, {"Content-Type": "application/json"}, JSON.stringify({speaker: speaker, presentations: speakerPresentations})];
    });

    this.delete('/api/speakers/:id', function(request) {
      // Do stuff
      return [200, {"Content-Type": 'application/json'}, '{}'];
    });

    this.get('/api/presentations', function (request) {
      return [200, {"Content-Type": "application/json"}, JSON.stringify({speakers: speakers, presentations: presentations})];
    });

    this.get('/api/presentations/:id', function (request) {
      // Do stuff
      return [200, {"Content-Type": "application/json"}, JSON.stringify({speaker: presentationSpeaker, presentation: presentation})];
    });
  });
}
```

**Add a**

**Test**

# Welcome to Boston Ember

[Home](#) [About](#) [Speakers](#) [Presentations](#)

## Presentation

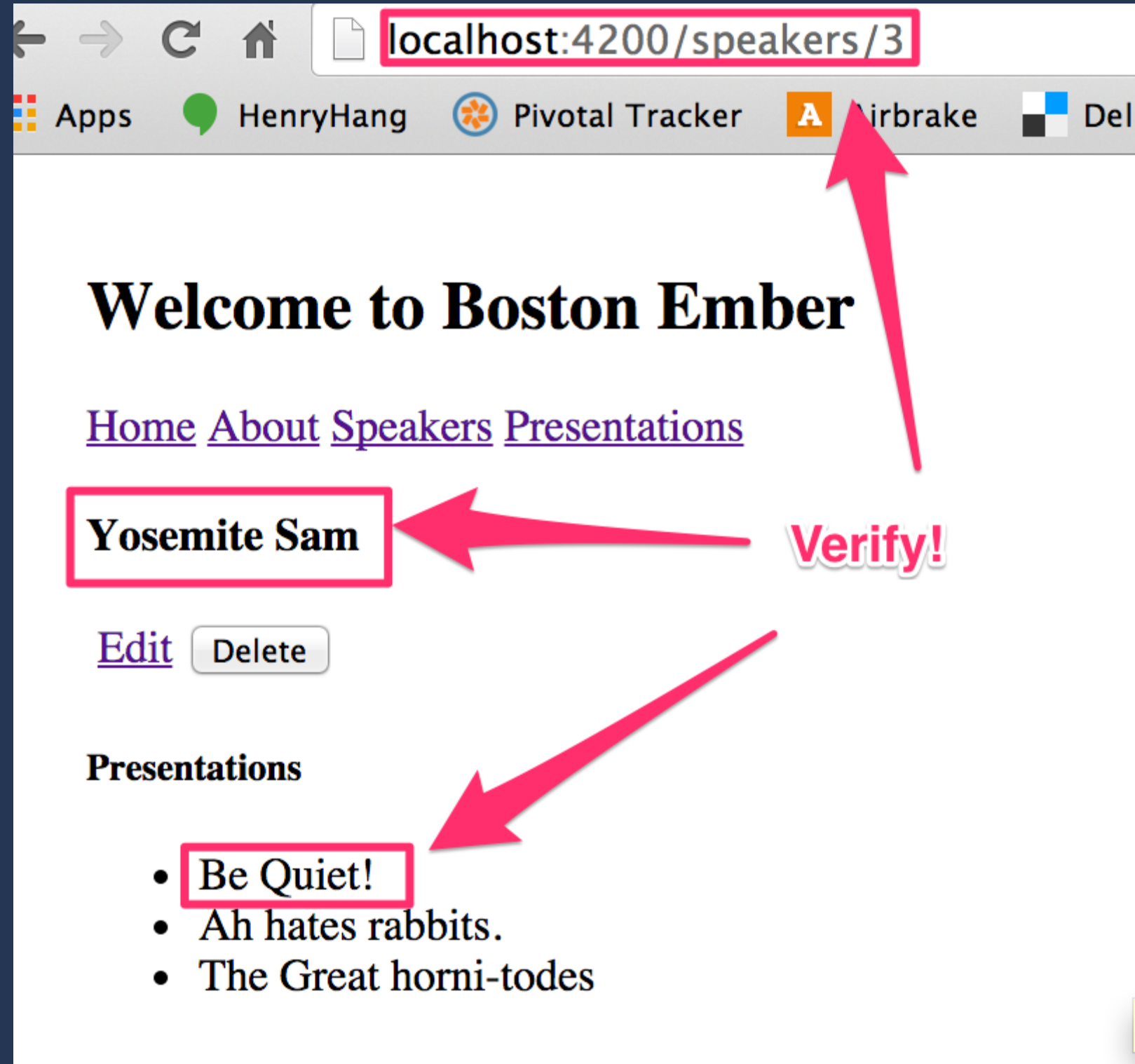
Speaker: Yosemite Sam

Edit Title:

Change to 'Be Quiet!'

Submit





# Set up the test

```
test('Should go to speaker\'s show page after submitting presentation edit', function() {  
    expect(0);  
});
```

# Set up the stimulus

```
test('Should go to speaker\'s show page after submitting presentation edit', function() {  
  expect(0);  
  visit('/presentations/4/edit');  
  var changed_text = 'Be Quiet!';  
  fillIn('input.presentation-title', changed_text);  
  click('button.commit-presentation-change');  
  andThen(function() {  
    // Gonna put assertions here.  
  });  
});
```

# Verify Response

```
test('Should go to speaker\'s show page after submitting presentation edit', function() {
  expect(3);
  visit('/presentations/4/edit');
  var changed_text = 'Be Quiet!';
  fillIn('input.presentation-title', changed_text);
  click('button.commit-presentation-change');
  andThen(function() {
    equal(currentURL(), '/speakers/3');
    ok("h4:contains('Yosemite Sam')");
    ok("li:contains('" + changed_text + "')");
  });
});
```