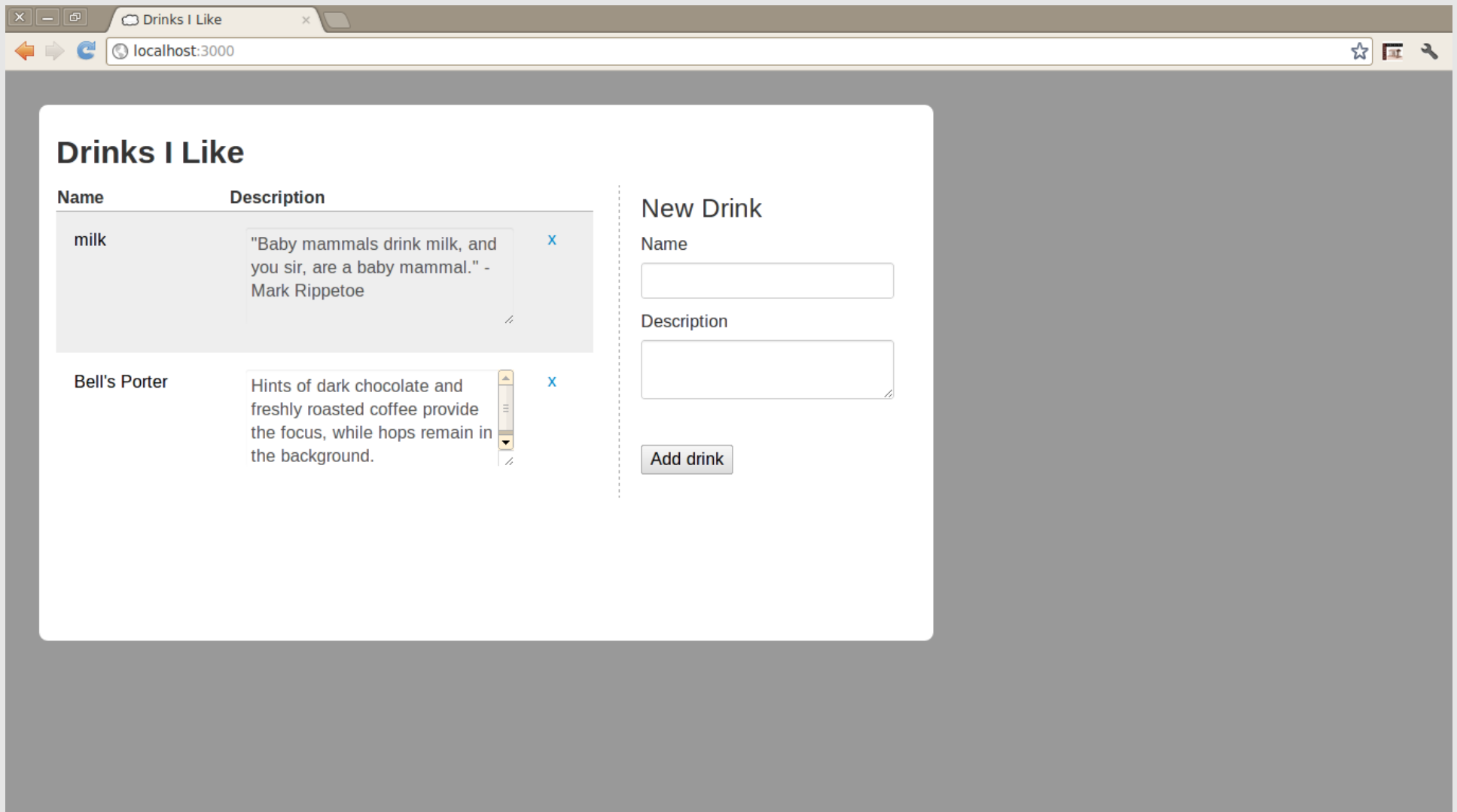
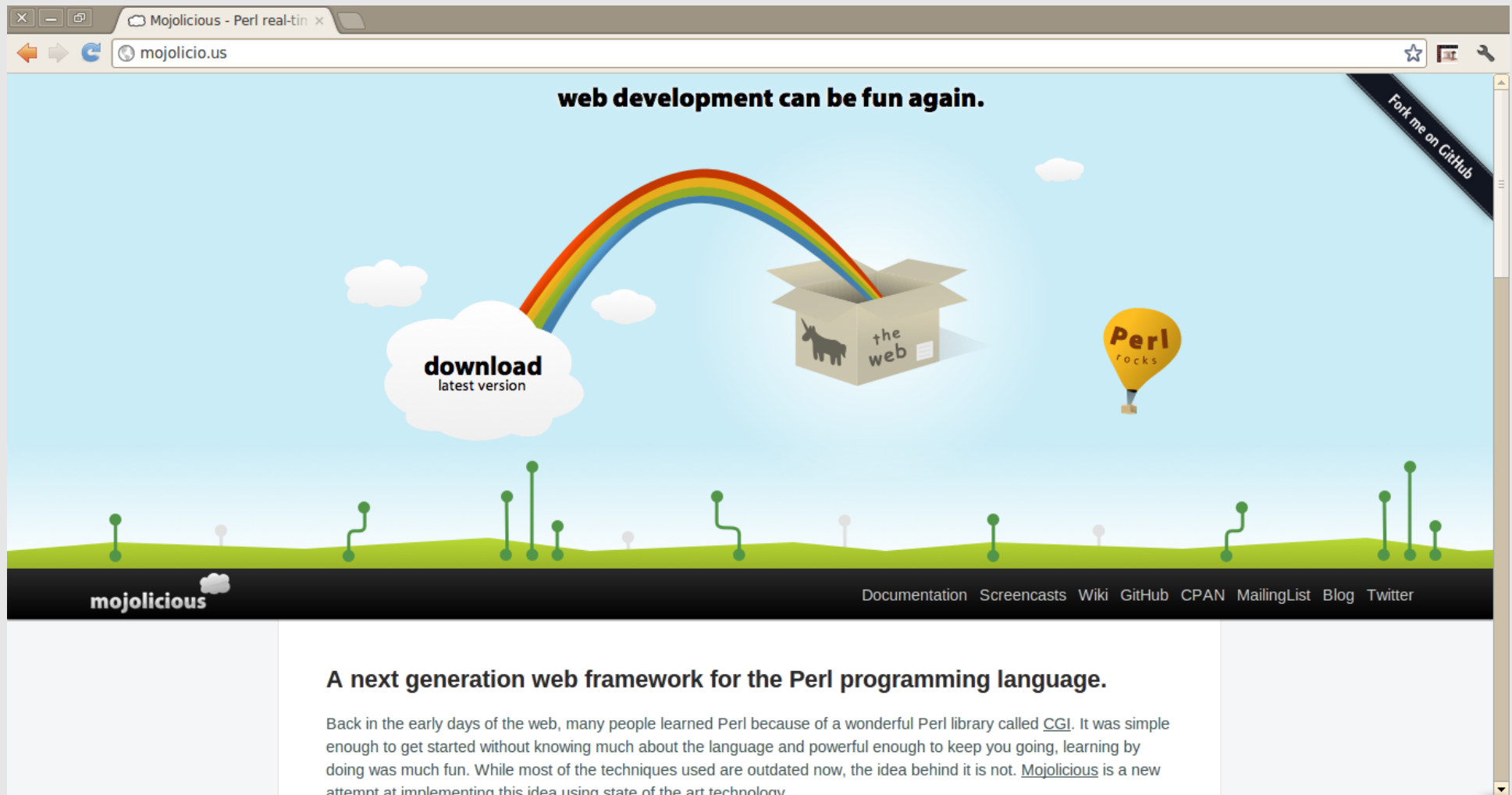


Single page webapps with
Mojolicious and Backbone.js

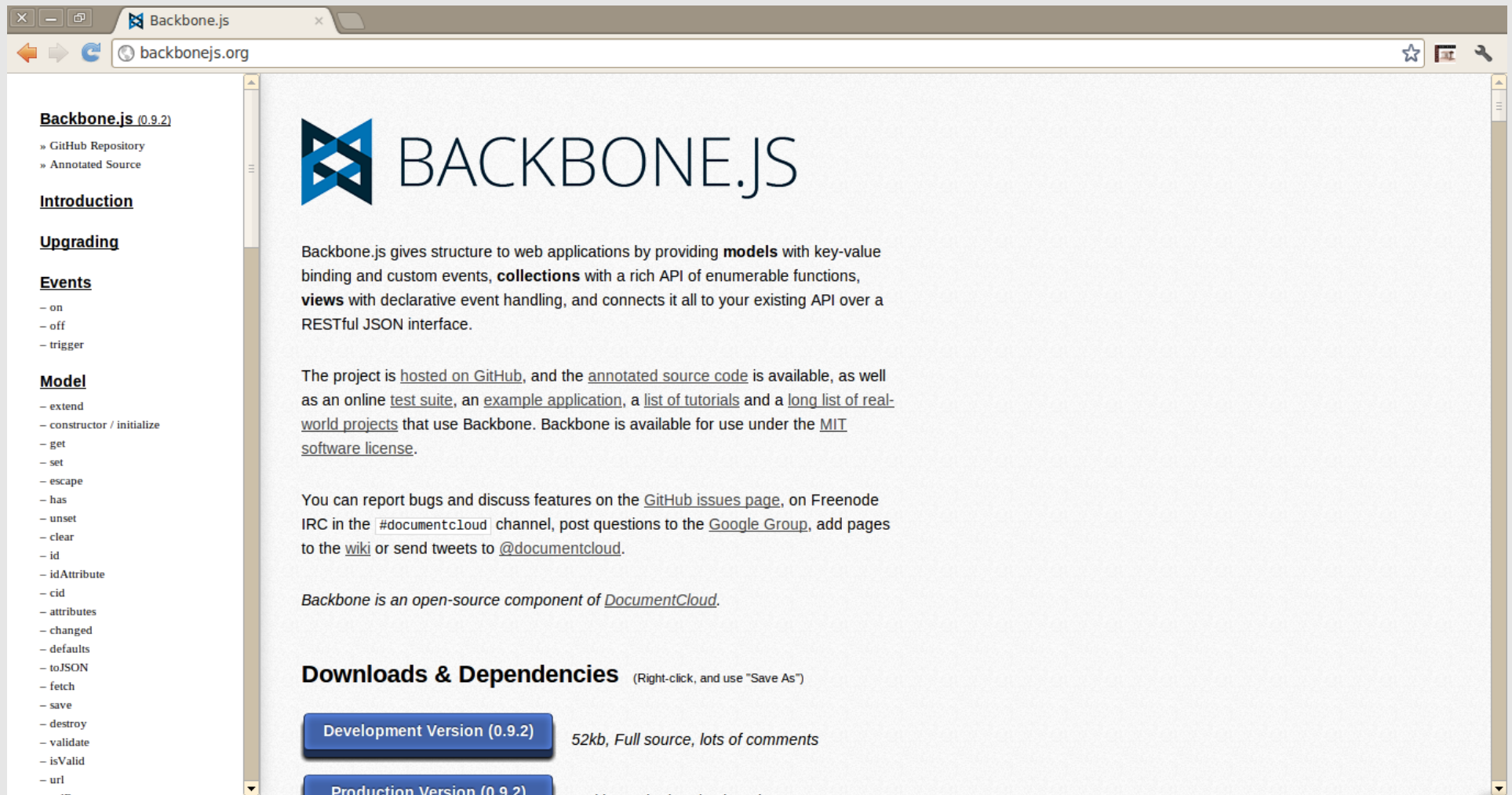
Here's the app...



Mojolicious, Backbone.js & Bootstrap



Mojolicious, Backbone.js & Bootstrap



The screenshot shows a web browser window with the address bar displaying "backbonejs.org". The page features the Backbone.js logo, which consists of a blue geometric shape resembling a stylized 'B' or a network node, followed by the text "BACKBONE.JS".

Backbone.js (0.9.2)

- » [GitHub Repository](#)
- » [Annotated Source](#)

Introduction

Upgrading

Events

- on
- off
- trigger

Model

- extend
- constructor / initialize
- get
- set
- escape
- has
- unset
- clear
- id
- idAttribute
- cid
- attributes
- changed
- defaults
- toJSON
- fetch
- save
- destroy
- validate
- isValid
- url

Backbone.js gives structure to web applications by providing **models** with key-value binding and custom events, **collections** with a rich API of enumerable functions, **views** with declarative event handling, and connects it all to your existing API over a RESTful JSON interface.

The project is [hosted on GitHub](#), and the [annotated source code](#) is available, as well as an online [test suite](#), an [example application](#), a [list of tutorials](#) and a [long list of real-world projects](#) that use Backbone. Backbone is available for use under the [MIT software license](#).

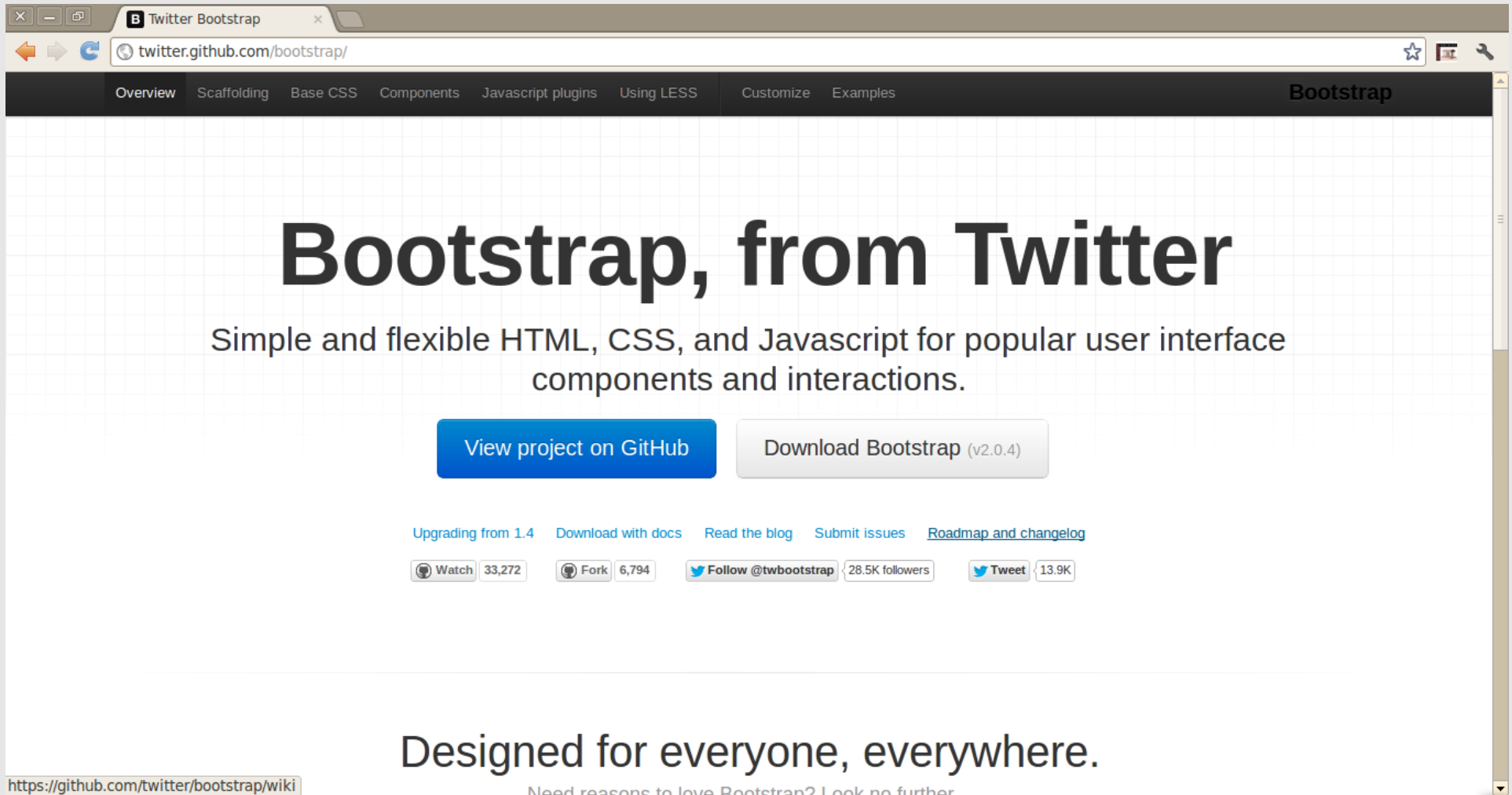
You can report bugs and discuss features on the [GitHub issues page](#), on Freenode IRC in the [#documentcloud](#) channel, post questions to the [Google Group](#), add pages to the [wiki](#) or send tweets to [@documentcloud](#).

Backbone is an open-source component of [DocumentCloud](#).

Downloads & Dependencies (Right-click, and use "Save As")

Version	Size / Description
Development Version (0.9.2)	52kb, Full source, lots of comments
Production Version (0.9.2)	

Mojolicious, Backbone.js & Bootstrap



A screenshot of a web browser displaying the Twitter Bootstrap GitHub repository page. The browser's address bar shows the URL `twitter.github.com/bootstrap/`. The page features a dark navigation bar with links for Overview, Scaffolding, Base CSS, Components, Javascript plugins, Using LESS, Customize, and Examples. The main content area has a light gray grid background. The title "Bootstrap, from Twitter" is prominently displayed in a large, bold, black font. Below the title, a subtitle reads "Simple and flexible HTML, CSS, and Javascript for popular user interface components and interactions." Two buttons are present: a blue "View project on GitHub" button and a gray "Download Bootstrap (v2.0.4)" button. A row of links includes "Upgrading from 1.4", "Download with docs", "Read the blog", "Submit issues", and "Roadmap and changelog". Social media and repository statistics are shown below, including "Watch 33,272", "Fork 6,794", "Follow @twbootstrap 28.5K followers", and "Tweet 13.9K". At the bottom, the text "Designed for everyone, everywhere." is displayed, followed by a partially visible line "Need reasons to love Bootstrap? Look no further".

Twitter Bootstrap

twitter.github.com/bootstrap/

Overview Scaffolding Base CSS Components Javascript plugins Using LESS Customize Examples Bootstrap

Bootstrap, from Twitter

Simple and flexible HTML, CSS, and Javascript for popular user interface components and interactions.

[View project on GitHub](#) [Download Bootstrap \(v2.0.4\)](#)

[Upgrading from 1.4](#) [Download with docs](#) [Read the blog](#) [Submit issues](#) [Roadmap and changelog](#)

[Watch](#) 33,272 [Fork](#) 6,794 [Follow @twbootstrap](#) 28.5K followers [Tweet](#) 13.9K

Designed for everyone, everywhere.

Need reasons to love Bootstrap? Look no further.

<https://github.com/twitter/bootstrap/wiki>

REST

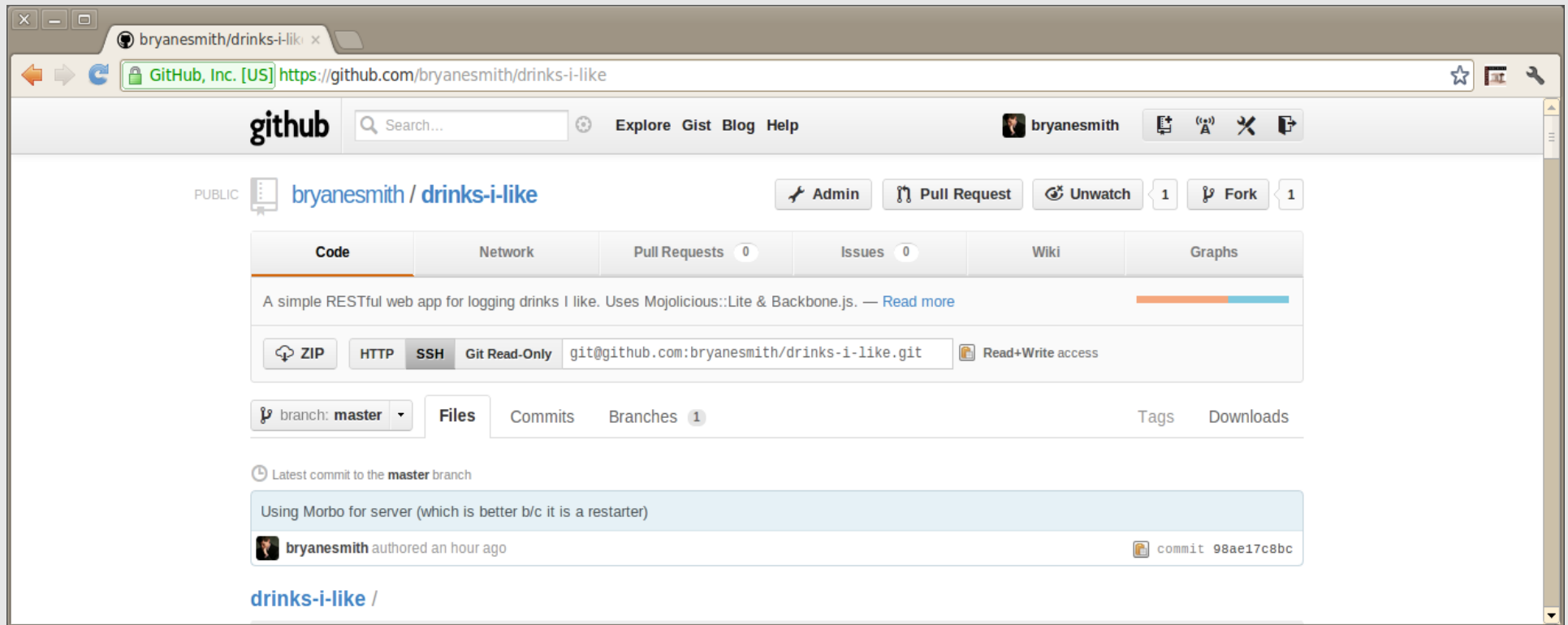
- Web API
- Data formats (e.g., XML, JSON, YAML, etc.)
- HTTP method (“verbs”) + resource
 - POST -> CRUD
 - GET * -> CRUD
 - PUT * -> CRUD
 - DELETE * -> CRUD

* Should be idempotent

RESTful design

Method	Route
POST	/api/drink (JSON object in body)
GET	/api/drink/ (Return all drinks)
	/api/drink/:id (Return one drink)
PUT	/api/drink/:id (JSON obj in body)
DELETE	/api/drink/:id

Let's install from GitHub



```
git clone git@github.com:bryanessmith/drinks-i-like.git
```


Mojolicious

```
38 # - - - - -
39 # Routes
40 # - - - - -
41
42 # Home page
43 get '/' => \&handle_home;
44
45 # Add a drink
46 post '/api/drink' => \&handle_post_drink;
47
48 # Get all drinks
49 get '/api/drink' => \&handle_get_drinks;
50
51 # Get drink
52 get '/api/drink/:id' => \&handle_get_drink;
53
54 # Update a drink
55 put '/api/drink/:id' => \&handle_put_drink;
56
57 # Delete a drink
58 del '/api/drink/:id' => \&handle_delete_drink;
59 del '/api/drink' => \&handle_delete_drink;
60
61 app->start;
```

Backbone.js: Model, Collection, View & misc

public/js/library.js:

```
$(document).ready( function() {  
    ...  
} );
```

Backbone.js: Model, Collection, View & misc

```
var Drink = Backbone.Model.extend({  
  urlRoot: '/api/drink'  
});
```

E.g.,

```
var drink = new Drink({title: 'Vietnamese iced  
coffee', description: '...' });
```

Backbone.js: Model, Collection, View & misc

```
var DrinkCollection = Backbone.Collection.extend({  
  model: Drink,  
  url: '/api/drink'  
});
```

```
var drinks = new DrinkCollection();
```

E.g.,

```
drinks.fetch();
```

```
drinks.add(drink);
```

Backbone.js: Model, Collection, View & misc

```
var DrinksView = Backbone.View.extend({  
  el: $('#drinks'),  
  templateId: '#drinks-template',  
  events: {  
    'click .remove-drink' : 'removeDrink',  
    'change .title' : 'updateTitle',  
    'change .description' : 'updateDescription'  
  },  
  ...  
});
```

Backbone.js: Model, Collection, **View** & misc

```
var DrinksView = Backbone.View.extend({  
  ...  
  initialize: function() {  
    // Bind events to collection  
    this.collection.bind('add', this.render, this);  
    this.collection.bind('remove', this.render, this);  
    // Grab template  
    this.template = $.template(this.templateId);  
    // Render  
    this.render();  
  },  
  ...  
});
```

Backbone.js: Model, Collection, **View** & misc

```
var DrinksView = Backbone.View.extend({  
  ...  
  render: function() {  
    var that = this;  
  
    $(this.el).find('td').remove();  
  
    // Render each drink  
    _(this.collection.models).each(function(drink) {  
      $.tmpl(that.template, drink.toJSON()).appendTo(that.el);  
    });  
  },  
  ...  
});
```

Backbone.js: Model, Collection, **View** & misc

- Add drink?

```
var drink = new Drink({title: title, description: desc });  
drink.save({wait: true});
```

- Update drink?

```
var drink.set(newValues);  
drink.save({wait: true});
```

- Delete drink?

```
drink.destroy({success: function(model, res) { coll.remove( drink ); }});
```

- Fetch drinks?

```
drink.fetch();
```


Backbone.js: Model, Collection, View & misc

```
// Grab drinks from server
```

```
drinks.fetch({  
  success: function() {  
    var drinksView = new DrinksView( { collection:  
drinks } );  
  }  
});
```

Backbone.js: Model, Collection, View & misc

```
// Handle add drink button
$('#add-drink').click(function () {
  // Create drink from form values
  var drink = new Drink({
    title: $('#add-title').val(),
    description: $('#add-description').val()
  });
  // Reset form
  $('#new-drink').each (function(){ this.reset(); });
  drink.save({wait: true});
  drink.fetch({
    success: function() { drinks.add(drink); }
  });
});
```

Awesome

- Asynchronous HTTP > page reload
- RESTful design keeps API simple
- Backbone.js makes it ~~easy~~ easier
- Bootstrap makes it pretty
- Mojolicious makes it fun

Needs improvement

- Delete should be idempotent
- Standard urls & Backbone.Router (History API)
- OO design improvements (e.g., fetching & helper functions should be part of view obj)
- More idiomatic Backbone.js

Resources

- [Mojolicious](#)
- [Backbone.js](#)
- [Bootstrap](#)
- [drinks-i-link on GitHub](#)
- [Designing a RESTful Web Application](#)

About me

- bryanesmith.com