Beginning



GoCode Thessaly #2

27 Sep 2013

Rich Internet App

```
$('form').submit(function(e){
  var errors = [];
  if ($('[name=firstName]', e.target).val() == '')
   errors.push('First name is required!');
  if ($('[name=age]', e.target).val() == '')
   errors.push('Age is required!');
  if (!$('[name=age]', e.target).val().match(/^d+$/))
   errors.push('Age must be a number!');
  if (errors.length == 0) {
   $.post('/posts',
     $(e.target).serialize(),
      function(data, status, xhr) {
       $('table > tbody').append('...');
       $('.modal').modal('close');
 } else {
   alert(errors.join(', '));
});
```

Rich Internet App

```
$('form').submit(function(e){
  var errors = [];
  if ($('[name=firstName
    errors.push('Fir
  if ($('[name=age
                                .val() ==
    errors.push(
                             .red!');
  if (!$('[name=
                           arget).val().mat
    errors.push
                           be a number!'
  if (errors.l
    $.post('/p
      $(e.targ
                        lize(),
      function
                        itus, x
                                         .d>...</t
        $( 'tab
        $('.mod
  } else {
    alert(errors.join
});
```

Enter backbone.js

"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."

http://backbonejs.org

Enter backbone.js

models collections views

routers events

history sync

Backbone.Model

```
var Note = Backbone.Model.extend({
   defaults: {
     color: 'transparent'
   },
   validate: function(attrs) {
     var errors = [];
     if (attrs.title == '') {
       errors.push("Title can't be blank!");
     }
     if (attrs.text == '') {
       errors.push("Body can't be blank!");
     }
     if (errors.length > 0) {
       return errors;
```

Backbone.Model (cont.)

```
var Sidebar = Backbone.Model.extend({
  promptColor: function() {
    var cssColor = prompt("Please enter a CSS color:");
    this.set({color: cssColor});
});
window.sidebar = new Sidebar;
sidebar.on('change:color', function(model, color) {
  $('#sidebar').css({background: color});
});
sidebar.set({color: 'white'});
sidebar.promptColor();
```

Backbone.Collection

```
var NotesCollection = Backbone.Collection.extend({
   model: Note,
  url: '/notes'
 });
 var Library = Backbone.Collection.extend({
   model: function(attrs, options) {
     if (condition) {
       return new PublicDocument(attrs, options);
     } else {
       return new PrivateDocument(attrs, options);
```

Backbone.View

```
var NoteView = Backbone.View.extend({
    tpl: Mustache.compile($('#noteTpl').html()),
    events: {
      'click .edit': function(e) {
        appRouter.navigate('/notes/' + this.model.get('id') + '/edit',
{trigger: true});
     },
      'click .delete': function(e) {
        this.model.destroy();
      }
   initialize: function() {
      this.listenTo(this.model, 'change', this.render);
      this.listenTo(this.model, 'destroy', this.remove);
   },
    render: function() {
      this.$el.html(this.tpl(this.model.attributes));
      return this;
   },
  });
```

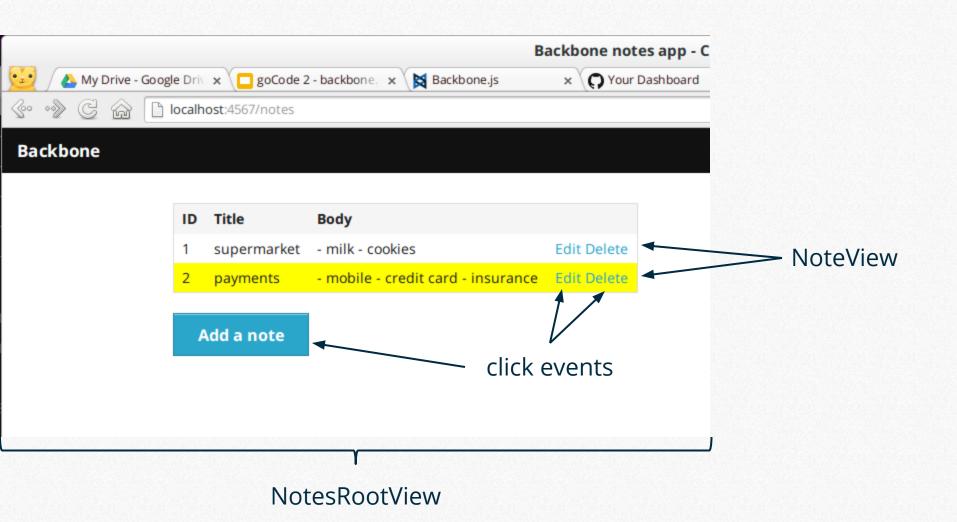
Backbone.Router

```
var NotesRouter = Backbone.Router.extend({
   routes: {
     'notes': 'notesIndex',
     'notes/:id/edit': 'notesEdit'
   },
   notesIndex: function() {
     $('.reveal-modal.open').foundation('reveal', 'close');
  },
   notesEdit: function(id) {
     new NoteEditView({
       model: notes.get(id)
     });
```

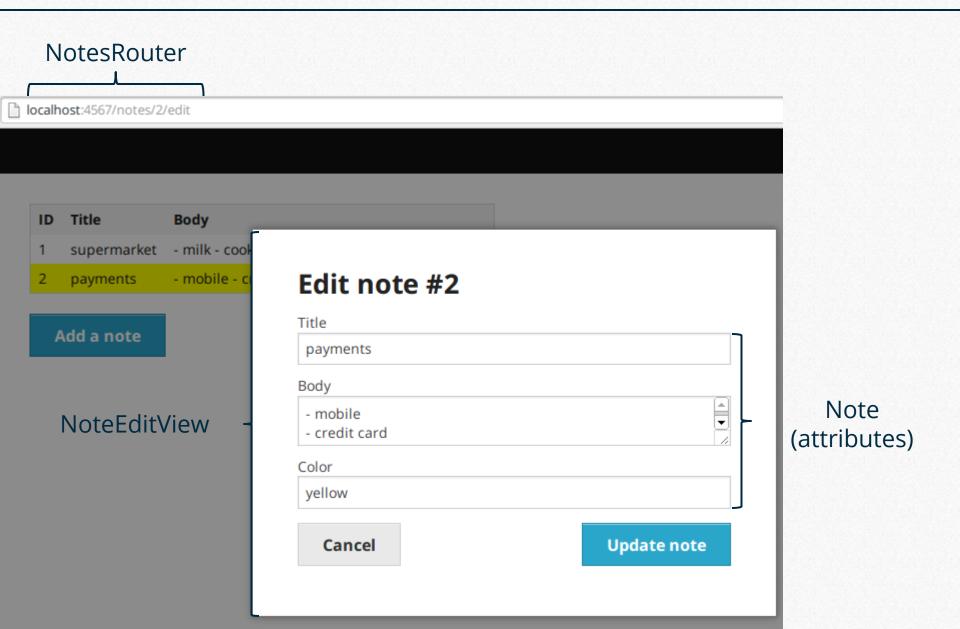
Backbone.Events

```
var ticket = {};
_.extend(object, Backbone.Events);
ticket.on("payment", function(gig) {
  alert("Paid for ticket " + gig.name);
});
ticket.trigger("payment", {name: "Metallica gig"});
book.on({
  "change:title": titleView.update,
  "change:author": authorPane.update,
  "destroy": bookView.remove
});
```

Overview of demo app



Overview of demo app (cont.)



Backbone vs MV(W|VM|C)







- → Fully featured framework vs library
- → Opinionated vs open
- → Rigid structure vs free-form
- → Long-lived apps vs short-lived pages

Demo time!



Thanks!



GoCode Thessaly

Code	https://github.com/GoCodeThessaly/code-2
Github org.	https://github.com/GoCodeThessaly
FB group	https://www.facebook.com/groups/goCode/
Google group	https://groups.google.com/forum/#!forum/gocode-thessaly