Backbone JS

BigLeap Solutions
gain momentum

http://bigleap.co.in

Introduction

- Name
- Technical background
- Experience & Expectations

Introduction

- Anilkumar GT
  - Co-Founder and Partner : BigLeap Solutions
  - Certified Ethical Hacker (EC-CEH)
  - Wildlife and Landscape Photographer:
    - http://www.FleetingMoments.net
- Youtube Channel:
  - Channel Name -> BigLeap Solutions
  - Videos on Design Thinking

Objectives / Expectations

- Backbone JS - Why / What
- Building blocks
- Model
- Collection
- Views
- Events
- Router
- Foundation UI

Backbone JS

- Why do we need something like BJS ?

- Lightweight library for structuring JS code
- MV* Pattern
- Event driven and declarative event handling
- Rest API

- Good structured JS based front-ends
- SPA

Dependencies

- Hard dependency
  - Underscore JS

- DOM Manipulation / Rest Integration
  - jQuery

- Templating
  - Underscore
  - Handlebars

Demo / Exercise

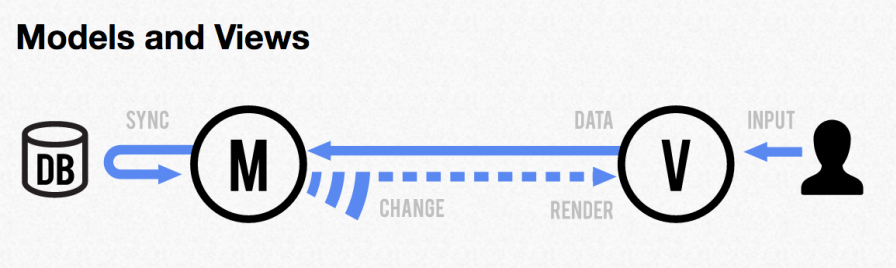
- Set up the boiler plate app
- Understand all the JS libraries and where to place them and how to use them

Building Blocks

- Models : Application Data
- Collections : Set of Models
- Views : Render models and handles events
- Events / Event Handling: Ability for components to Pub / Sub to each other
- Routers: View navigation
  - Not required if not an SPA

Models

- Domain objects
  - Attributes, Validation, Behaviours

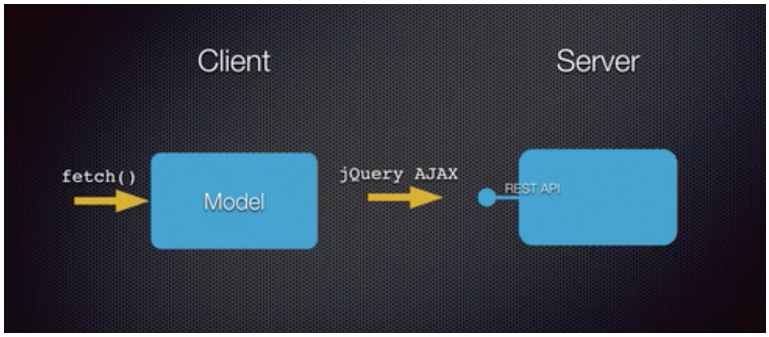


Demo / Exercise

- Create Backbone model:
  - Initialise
  - Defaults
  - Validate
  - set / unset / clear
  - isValid / validationError

Models

- Could sync with a Rest API
- Async / jQuery XHR
- Fetch -> GET
- Save -> POST / PUT
- Destroy -> DELETE



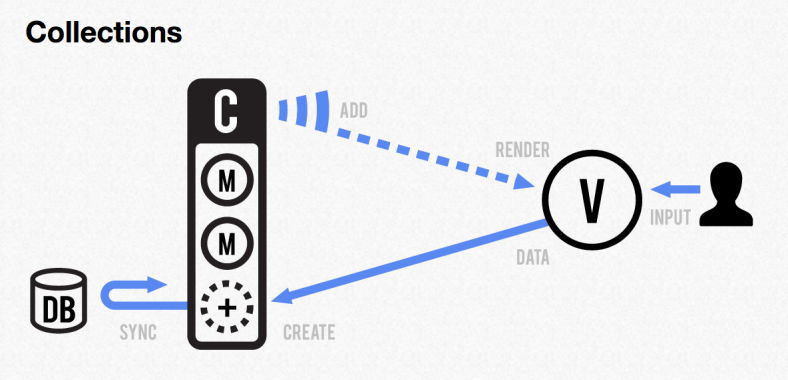
Demo / Exercise

- Get the /books Rest API working
- Test using RestClient

- Interact with the /books Rest API
- Success / Failure callbacks

Collections

- Ordered sets of models
- Could interact with Rest API
- Fetch -> GET



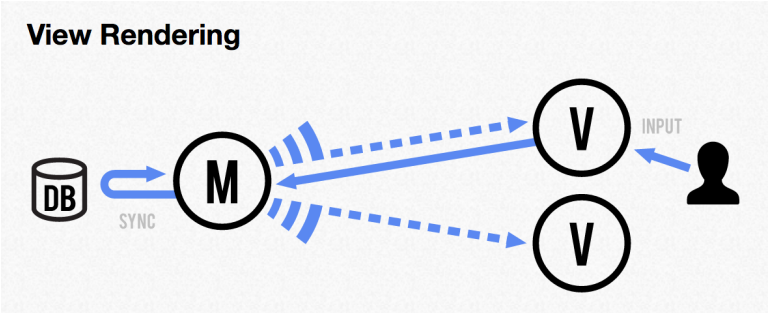
Demo / Exercise

- Backbone Collection
- add / push / add-at / unshift
- remove
- _ where and fromWhere
- _ filter

- Interact with Rest API

View

- Handel user interactions
- Backed by Model or Collections
- Handle events
- Most crucial aspect to get right -> Partial View updates / Page updates



Demo / Exercise

- View
- View Elements and Rendering
- Passing Model to view

- Handling model events
- Handling collections events

Collections

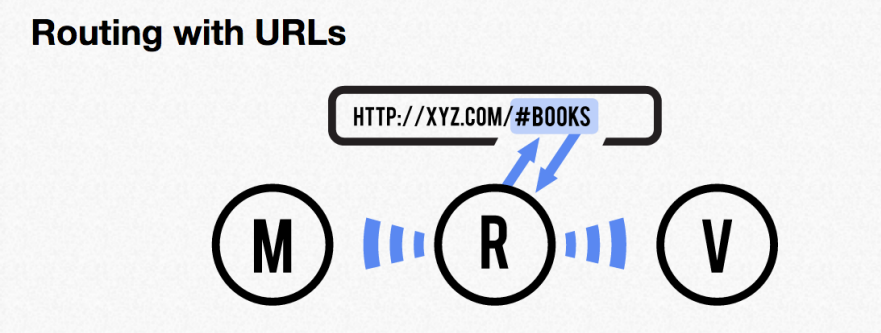
- Integrate with HTML
- Got to be extremely careful with design choices
- Do not break DRY / SOC
- Underscore option
- Handlebars

Demo / Exercise

- Underscore Template
  - How not to use templates
- Handlebars Template
- Handling DOM Events

Router

- Methods for routing client-side pages, and connecting them to actions and events
- Needed if the application is a SPA
- Could use /page or #fragments based routes
- Don't forget the 'start' method



Demo / Exercise

- Create 3 Views and Router to navigate

Foundation UI

- Responsive Design Framework
  - Sites / Email / Apps
- Latest v6.0
- Why would we need some thing like this ?

- CSS / SASS / Angular based components
- Improves front-end engineering
  - Setup, Development environments, automation
  - Panini -> Handlebars
  - JS / SASS / Images

Demo / Exercise

- Setting up Zurb Foundation CLI

- Setting up a site with zurb template
- Setting up the entire dev stack and running the stack
- Writing a new zurb template
- Writing a new fragment

References

- Backbone JS : http://backbonejs.org
- Underscore JS : http://underscorejs.org
- Handlebar JS : http://handlebarsjs.com/
- Curb Foundation : http://foundation.zurb.com/

- Marionette JS : http://marionettejs.com/

Thank You !!!

BigLeap Solutions
gain momentum

http://bigleap.co.in