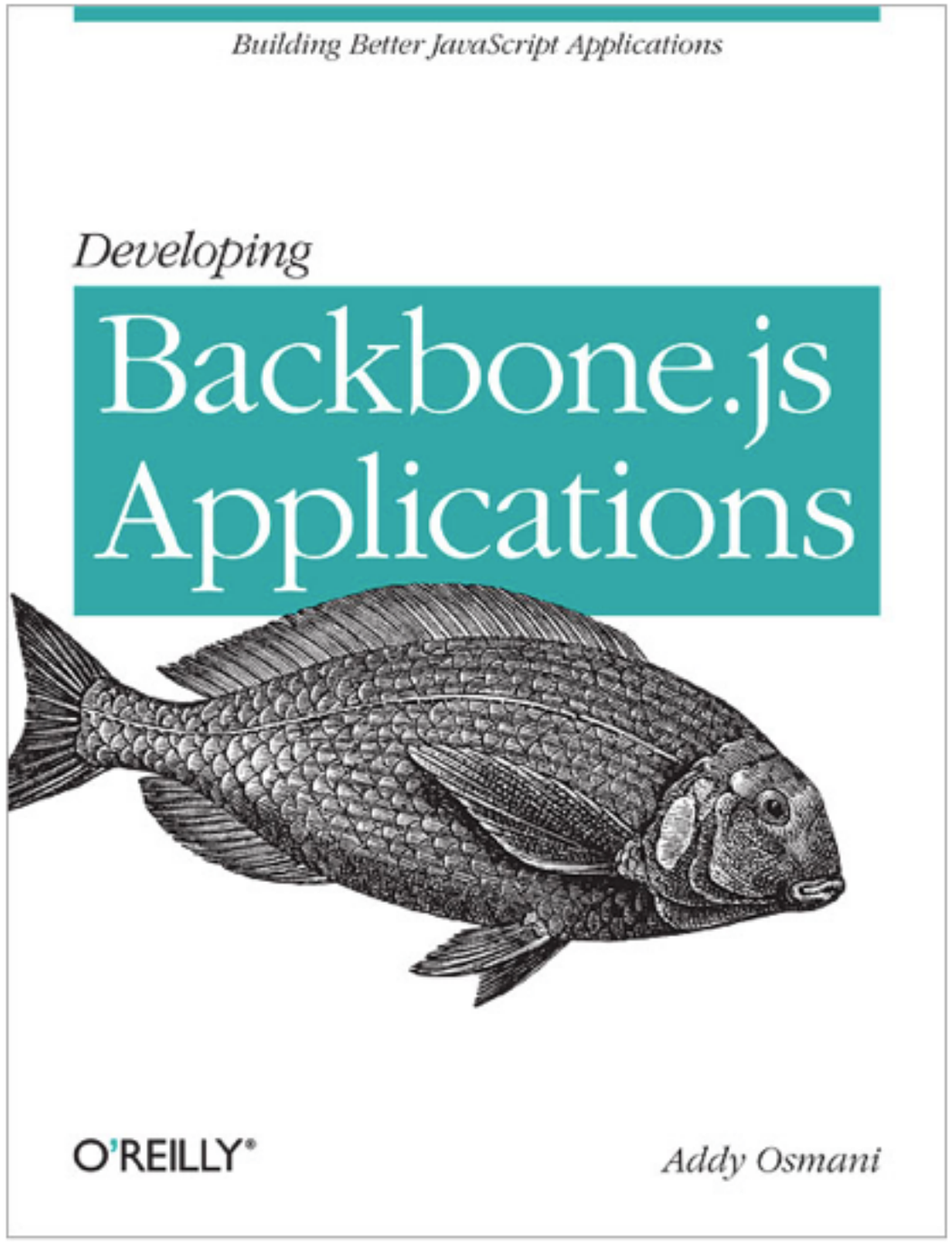
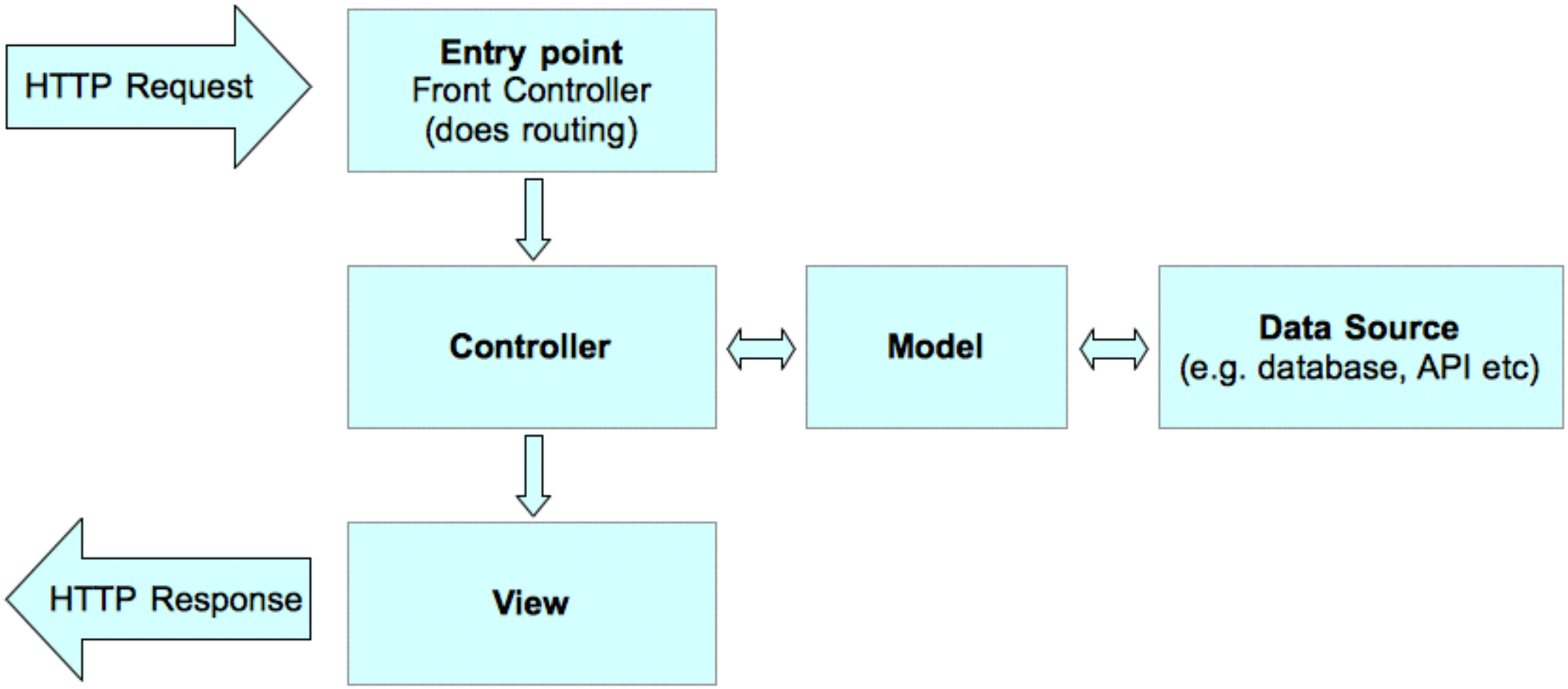
serious #**js**

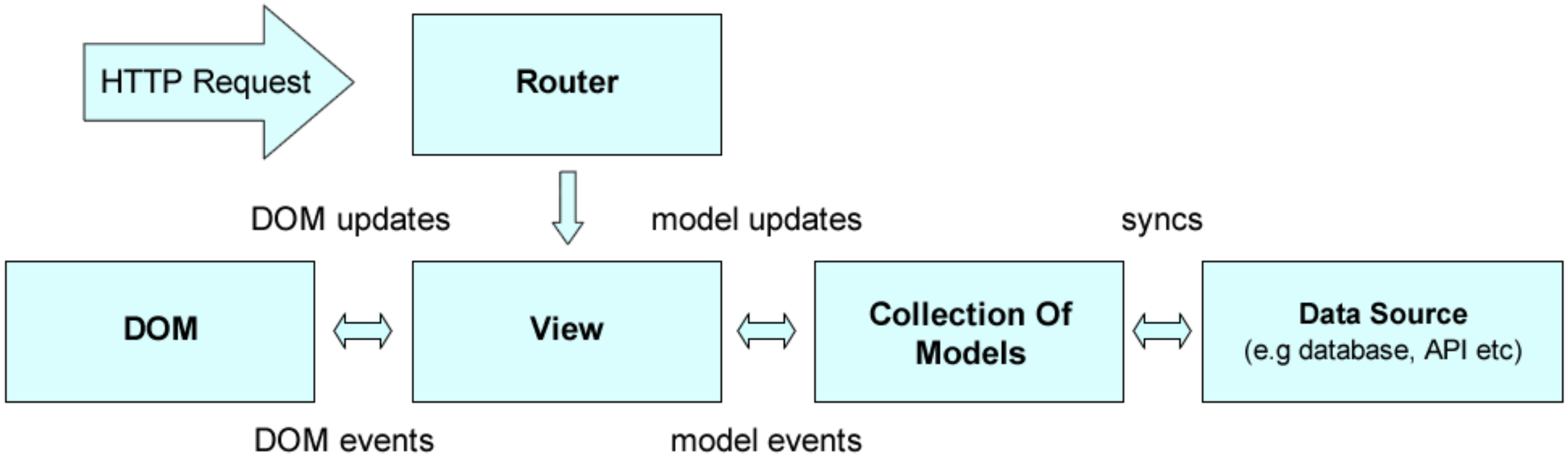
to **infinity** and beyond!

plaatje: http://www.disneypictures.tk/disney-pictures/buzz-lightyear-pictures.html



http://addyosmani.github.io/backbone-fundamentals/


HTTP Request
Entry point
Front Controller
(does routing)
Controller
Model
Data Source
(e.g. database, API etc)
View
HTTP Response


HTTP Request
Router
DOM updates
model updates
syncs
DOM
View
Collection Of Models
Data Source
(e.g database, API etc)
DOM events
model events

backbone model

assignment

Gegeven is **Match** met de volgende attributen:
homeMatch, homeClub, awayClub, stadium, city.

Maak van **Match** een model in backbone.

readings

model

http://backbonetutorials.com/what-is-a-model/

https://courses.tutsplus.com/courses/connected-to-the-backbone/
(vids 1.1 - 2.3)

backbone model view

assignment

Gegeven is model **Match** met de volgende attributen:
homeMatch, homeClub, awayClub, stadium, city.

Maak van het model **Match** een **view** in backbone.

assignment

Gegeven is model **Match** met de volgende attributen:
homeMatch, homeClub, awayClub, stadium, city.

Maak binnen de view gebruik van een template die een li rendered.

readings

view / template

http://backbonetutorials.com/what-is-a-view/

https://courses.tutsplus.com/courses/connected-to-the-backbone/
(vids 2.4 - 2.6)

backbone model view event

assignment

Gegeven is model **Match** met de volgende attributen:
homeMatch, homeClub, awayClub, stadium, city.

Maak een click event in de view van het model Match
die de achtergrond kleur van het li element toggled.

readings

events

http://backbonetutorials.com/what-is-a-view/

https://courses.tutsplus.com/courses/connected-to-the-backbone/
(vids 3.1 - 3.4)

backbone collection

assignment

Gegeven is model **Match** met de volgende attributen: homeMatch, homeClub, awayClub, stadium, city.

Maak van het model **Match** een **collection** (Matches) in backbone. (hardcode json)

Probeer de collection van de url ‘http://docent.cmi.hr.nl/moora/imp03/api/wedstrijden' uit te lezen

readings

collection

http://backbonetutorials.com/what-is-a-collection/

https://courses.tutsplus.com/courses/connected-to-the-backbone/
(vids 2.7 - 2.10)

backbone collection view

assignment

Gegeven is model **Match** met de volgende attributen:
homeMatch, homeClub, awayClub, stadium, city.

Maak van de collection **Matches** een **view** in backbone.

*zorg er voor dat template en het click event uit de vorige opdracht blijven werken.

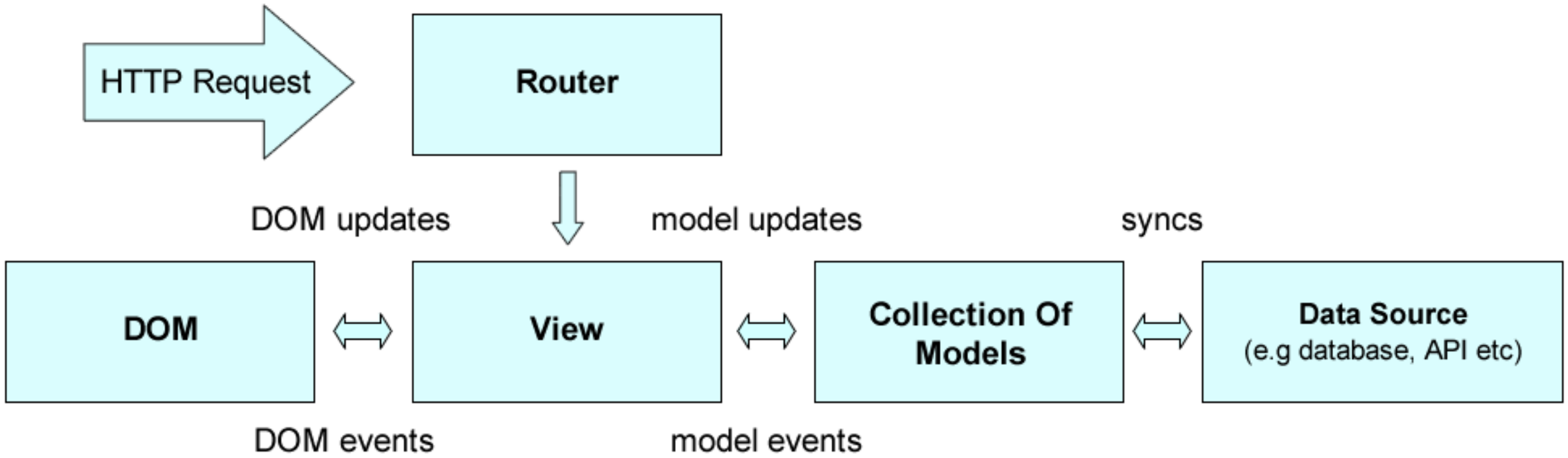
readings

collection

http://backbonetutorials.com/what-is-a-collection/

https://courses.tutsplus.com/courses/connected-to-the-backbone/
(vids 2.7 - 2.10)

backbone router



http://backbonetutorials.com/what-is-a-router/#/posts/130

Web applications often provide linkable, bookmarkable, shareable URLs for important locations in the app. Until recently, hash fragments (#page) were used to provide these permalinks, but with the arrival of the History API, it's now possible to use standard URLs (/page).

```
//Example router with custom routes
var Router = Backbone.Router.extend({
    routes: {
        'books/:book': ‘bookAction’,
        ‘test/:test/:more’: ‘testAction’ //Possible to create bigger urls
    },

    /**
     * Route callback
     *
     * @param book
     */
    bookAction: function (book) {
        //Do something with book.
    }
});
```

```
//Enables history for routing state
Backbone.history.start();
```

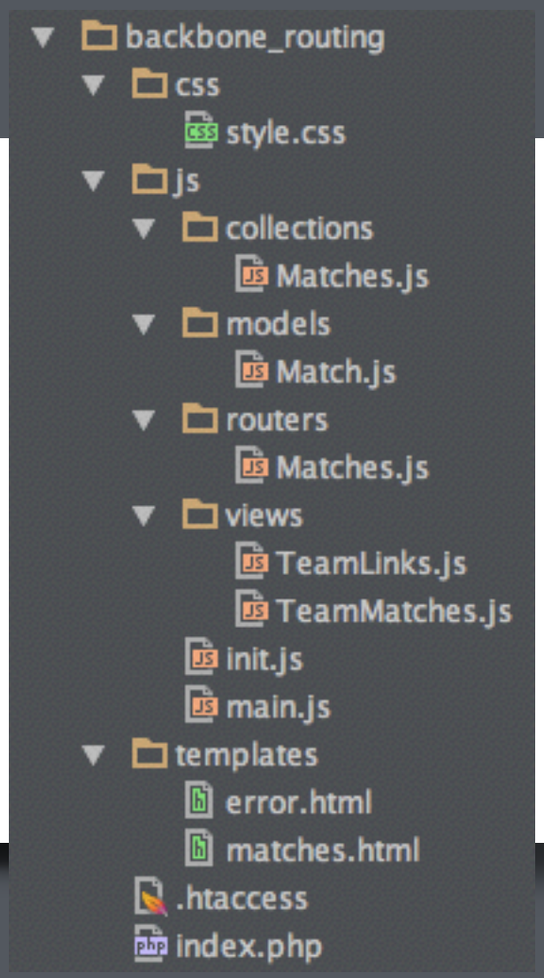
#JS - BACKBONE HISTORY

code

Note that using real URLs requires your web server to be able to correctly render those pages, so back-end changes are required as well. For example, if you have a route of /documents/100, your web server must be able to serve that page, if the browser visits that URL directly.

For full search-engine crawlability, it's best to have the server generate the complete HTML for the page ... but if it's a web application, just rendering the same content you would have for the root URL, and filling in the rest with Backbone Views and JavaScript works fine.

```
//init.js
(function () {
    window.site = {};
    site.$document = $(document);
    site.views = {};
    site.collections = {};
    site.models = {};
    site.routers = {};
    site.events = _.clone(Backbone.Events);
})();
```



#JS - JS STRUCTURE

code

assignment

Gegeven is model **Match** met de volgende attributen: homeMatch, homeClub, awayClub, stadium, city.

Breidt de MatchesListView zo uit dat op basis van de routing dynamisch de wedstrijden worden ingeladen.

- Routing op basis van **/matches/:league/:club**
- Haal **data** van het team op door de url te veranderen.
  - ~/ndex.html#league/PremierLeague/Arsenal
- **Toon matches** op scherm.

readings

router

http://backbonetutorials.com/what-is-a-router/

https://courses.tutsplus.com/courses/connected-to-the-backbone/
(vids 4.1 - 4.3)

huiswerk

- Oefenen met stof van Backbone lessen!
- **Werken aan eindopdracht**