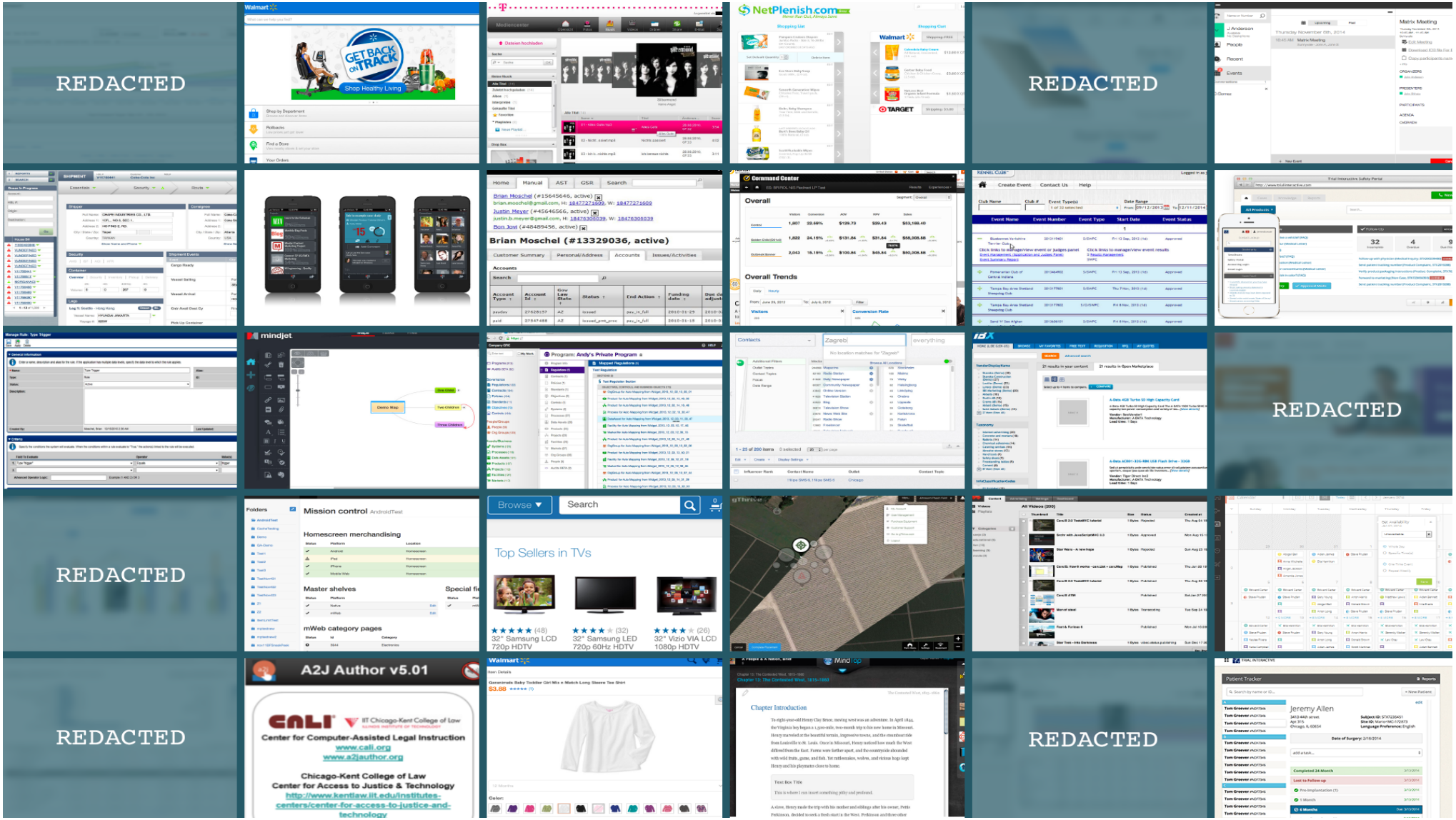
Building SPAs the “right” way

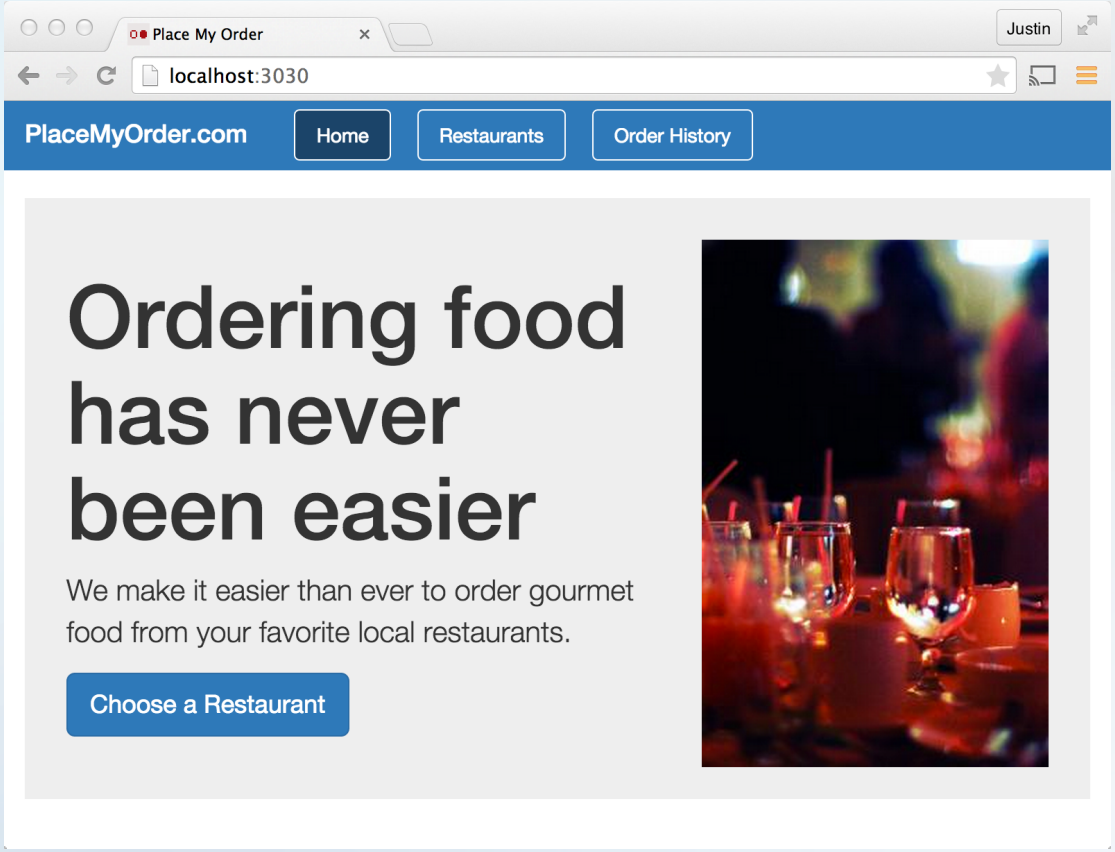
Making apps maintainable,
fun, and fast.


REDACTED
Walmart
GET BACK on TRACK
Shop Healthy Living
NetPlenish.com
TARGET
REDACTED
Brian Moschel (#13329036, active)
Overall
Overall Trends
REDACTED
mindjet
REDACTED
REDACTED
Mission control
Homescreen merchandising
Master shelves
mWeb category pages
Top Sellers in TVs
32" Samsung LCD 720p HDTV
32" Samsung LED 720p 60Hz HDTV
32" Vizio VIA LCD 1080p HDTV
REDACTED
A2J Author v5.01
CALI
IIT Chicago-Kent College of Law
Center for Computer-Assisted Legal Instruction
www.cali.org
www.a2jauthor.org
Chicago-Kent College of Law
Center for Access to Justice & Technology
http://www.kentlaw.iit.edu/institutes-centers/center-for-access-to-justice-and-technology
Walmart
Chapter Introduction
REDACTED
Patient Tracker
Jeremy Allen

Outline

1. Checklists
2. Modlets
3. Semver Packages
4. Custom HTML Composition
5. Unit Testable ViewModels
6. Two Way routing
7. SuperModels
8. Live Reload
9. Smart Progressive Loading
10. Server Side Rendering

place-my-order



github.com/bitovi/place-my-order

Outline

1. Checklists
2. Modlets
3. Semver Packages
4. Custom HTML Composition
5. Unit Testable ViewModels
6. Two Way routing
7. SuperModels
8. Live Reload
9. Progressive Loading
10. Server Side Rendering

1. Checklists

A JavaScript Project Checklist

http://bitovi.github.io/checklist/

https://github.com/bitovi/checklist

Management

- ✓ Release < 6 months 0.48
- ✓ Yearly Trainings 0.46
- ✓ Vision, Goals, Strategy, Roadmap 0.45
- ✓ Social Events 0.45

Design

- User testing 0.45
- Design documentation 0.44

Development

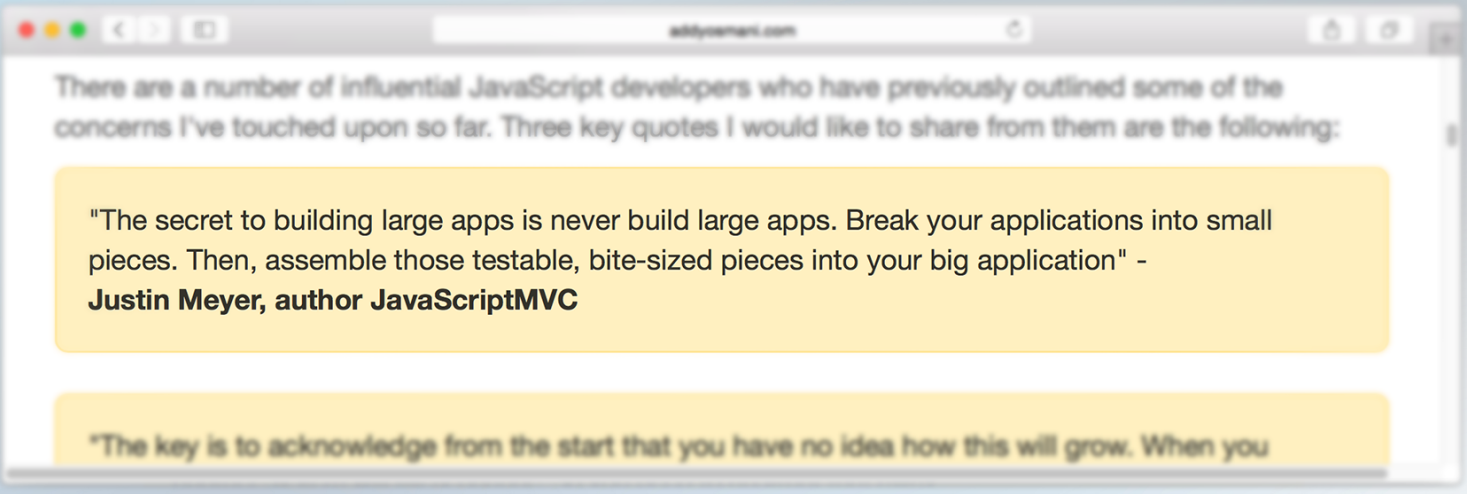
- Is every module Code Reviewed? 0.28
- Is every module Documented? 0.28
- Is every module Tested? 0.20

A JavaScript Project Checklist

http://bitovi.github.io/checklist/

https://github.com/bitovi/checklist

2. Modlet Workflows



Source: http://addyosmani.com/largescalejavascript/

FOLDERS

- project/
  - js/
    - moduleA.js
    - moduleB.js
  - templates/
    - moduleA.handlebars
    - moduleB.handlebars
  - css/
    - moduleA.css
    - moduleB.less
  - test/
    - moduleA_test.js
    - moduleB_test.js
  - docs/
    - moduleA.markdown
    - moduleB.markdown

FOLDERS

- project/
  - moduleA/
    - moduleA.js
    - moduleA.handlebars
    - moduleA.css
    - moduleA_test.js
    - moduleA.markdown
    - test.html
    - moduleA.html
  - moduleB/

<-- MODLETS!

Benefits

- Easier to identify missing tests or documentation.
- More likely to update tests and documentation.
- Good API design.
- Easier to develop in isolation.

Example

- http://localhost:3030/app/order/demo.html
- http://localhost:3030/app/order/test.html

3. Semver Packages

- Put modules in their own repository
- Publish them to NPM (or bower)
- Import and use in the main app

Place My Order

Justin

localhost:3030/restaurants/crab-barn/order

PlaceMyOrder.com

Home | Restaurants | Order History

Order from Crab Barn

Lunch menu | Dinner menu

2 selected

- [ ] Chicken with Tomato Carrot Chutney Sauce $45.99
- [x] Spinach Fennel Watercress Ravioli $35.99
- [ ] Truffle Noodles $14.99

Name:

Address:

Phone:

Total: $59.98

Place My Order!

javascript://

bitovi-components/bit-tab

Justin

GitHub, Inc. [US] https://github.com/bitovi-components/bit-tabs

This repository Search

Explore Gist Blog Help

bitovi-components / bit-tabs

A tabs component for CanJS — Edit

40 commits 3 branches 3 releases 4 contributors

branch: master bit-tabs / +

Make the tests more FuncUnit-y

matthewp authored 25 days ago latest commit 0a213ad03e

src	Allow passing ul class using tabsClass attr.	25 days ago
test	Make the tests more FuncUnit-y	25 days ago
.gitignore	build basics working	4 months ago
.npmignore	build basics working	4 months ago
.travis.yml	Adding Testee and Travis CI.	4 months ago
Gruntfile.js	0.0.3	a month ago
README.md	Fixing the readme	a month ago

public — bash — 92×28

```
Last login: Wed May  6 17:31:09 on ttys006
Justins-MacBook-Pro:place-my-order justin$ cd public/
Justins-MacBook-Pro:public justin$
```

order.js — place-my-order

FOLDERS

- place-my-order
  - bin
  - lib
  - node_modules
  - presentation
    - javascript.js
    - page.html
  - public
    - app
      - header
      - home
      - models
      - order
        - analytics
        - history
        - phone
        - demo.html
        - order.js
        - order.md
        - order.stache
        - order_test.js
        - test.html
      - restaurant
      - appstate.js
      - layout.stache
      - main.stache
      - routes.js
      - test.js
    - demo
    - images
    - less
    - node_modules

order.js | order.stache

```
1  import Component from 'can/component/component';
2  import Map from 'can/map/';
3  import 'can/map/define/';
4  import template from './order.stache!';
5  import Restaurant from 'app/models/restaurant';
6  import Order from 'app/models/order';
7
8  export const ViewModel = Map.extend({
9    define: {
10     /**
11      * @property {String} slug
12      *
13      * The restaurants slug (short name). Will
14      * be used to request the actual restaurant.
15      */
16     slug: {
17       type: 'string'
18     },
19     /**
20      * @property {app/models/order} order
21      *
22      * The order that is being processed. Will
23      * be an empty new order inititally.
24      */
25     order: {
```

Line 6, Column 38 | Spaces: 2 | JavaScript

Dependency Management Options

```
import bitTabs from 'bit-tabs';

// CommonJS / Browserify
var bitTabs = require('bit-tabs');

// AMD / r.js
define(['bit-tabs'], function(bitTabs){});

<!-- Script and Link tags -->
<script src='bit-tabs.js'></script>
<link href='bit-tabs.css' rel='stylesheet'>
```

bit-tabs.js — bit-tabs

FOLDERS

- bit-tabs
  - node_modules
  - src
    - bit-tabs.js
    - panel.stache
    - tabs.less
    - tabs.stache
    - util.js
  - test
  - .gitignore
  - .npmignore
  - .project
  - .travis.yml
  - bit-tabs.html
  - Gruntfile.js
  - package.json
  - README.md

```
1  import can from "can";
2  import stache from "can/view/stache/";
3  import util from "./util";
4  import tabsStache from "./tabs.stache!";
5  import panelStache from "./panel.stache!";
6  import "./tabs.less!";
7
8  export var BitPanelVM = can.Map.extend({
9      active: false
10 });
11
12 can.Component.extend({
13     tag:"bit-panel",
14     template: panelStache,
15     scope: BitPanelVM,
16     events: {
17         inserted: function(){
18             this.element.parent().scope().addPanel( this.scope );
19         },
20         removed: function(){
21             this.element.parent().scope().removePanel( this.scope );
22         }
23     }
24 });
25
26 export var BitTabsVM = can.Map.extend({
```

Line 1, Column 1 | Tab Size: 4 | JavaScript

Semver Packages Benefits

- Reuse
- Thinking Semver
- Easier Release Management
- Client Side Open Source

4. Custom HTML Composition

- It’s awesome!
- Composable Elements
- Template Dependencies
- Composing Custom HTML Elements

page.html

```
<html>
    <body>
        <input id='datepicker'/>
        <script src='jquery.js'></script>
        <script src='myapp.js'></script>
    </body>
</html>
```

myapp.js

```
$("#datepicker").datepicker();
```

react.jsx

```
render: function() {
  return <div>
    <DatePicker value="date" />
  </div>;
}
```

react.jsx ✖ angular.html ✖

```
<datepicker ng-model="date"/>
```

react.jsx ✖ angular.html ✖ canjs.stache ✖

```
<datepicker value="{date}"/>
```

“Tag team”

- elements that load data.

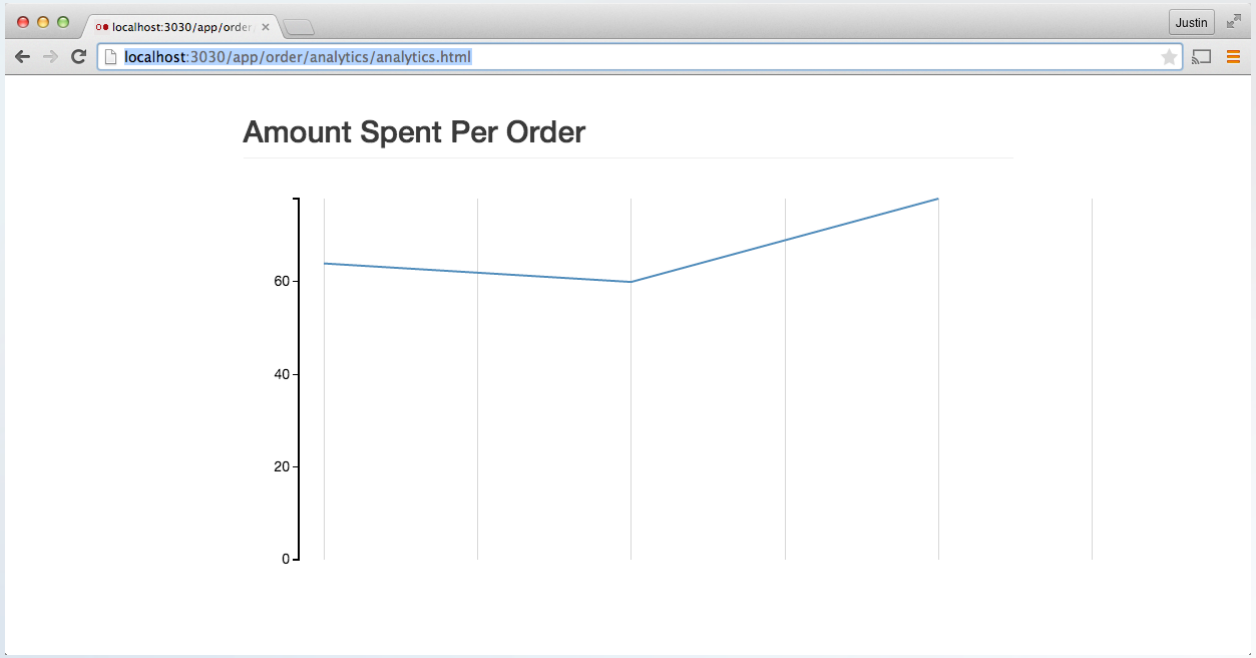
```
<order-model find-all="{status=‘new’}”/>
```

- composable widget elements.

```
<bit-graph title=”New Vs Old">
  <bit-series data="{newData}" />
  <bit-series data="{oldData}" />
</bit-graph>
```

DEMO

- http://localhost:3030/app/order/analytics/analytics.html



analytics.stache

```
<can-import from="app/models/order" />
<can-import from="ui/bit_graph/bit_graph"/>


<order-model findAll="{}" [orderdata]="{value}"/>

<bit-graph title="Analytics">
    <bit-series data="{../orderdata.totals}" />
</bit-graph>
```

example.jsx

```
var HelloWorld = React.createClass({
  displayName: 'hello-world',

  getInitialState: function() {
    return {
      message: "Hello"
    };
  },

  render: function() {
    return <div>{{message}} World!</div>;
  }
});
```

webcomponent.html

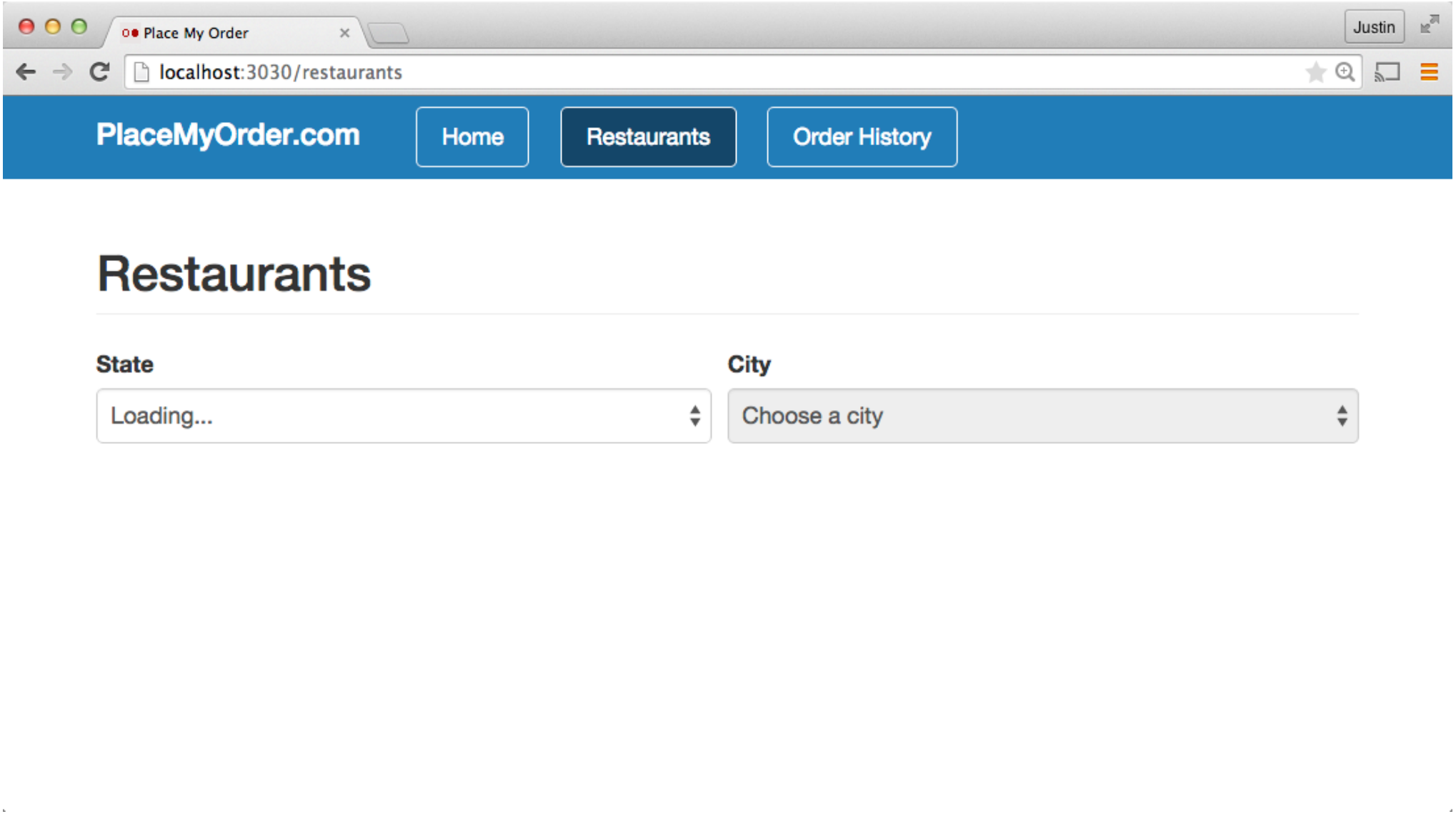
```
<template id="sdtemplate">
  <style> p { color: orange; }  </style>
  <p>Hello World</p>
</template>
<script>
var proto = Object.create(HTMLElement.prototype, {
  createdCallback: {
    value: function() {
      var t = document.querySelector('#sdtemplate');
      var clone = document.importNode(t.content, true);
      this.createShadowRoot().appendChild(clone);
    }
  }
});
document.registerElement(’hello-world', {prototype: proto});
</script>
```

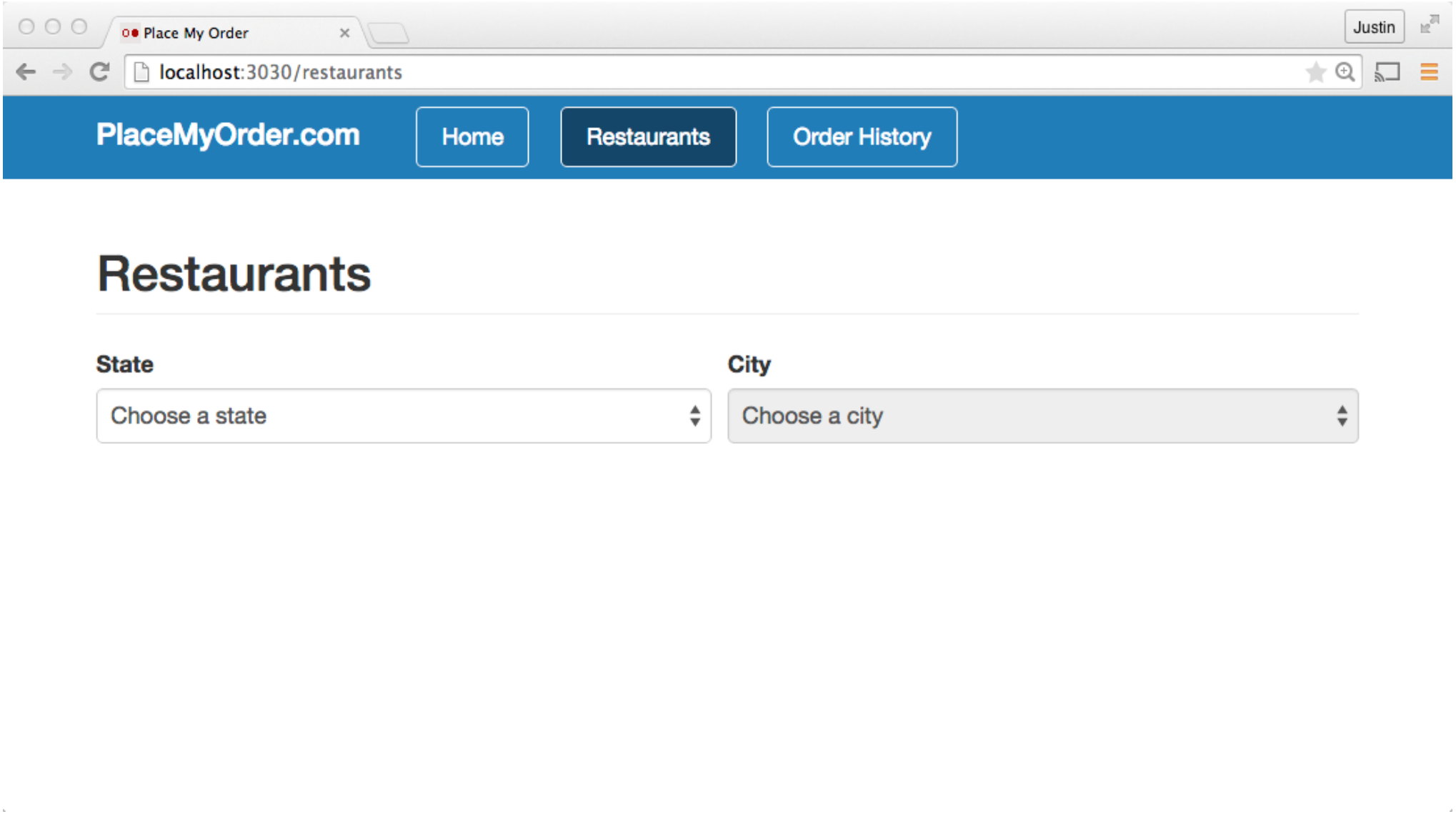
canjs.js

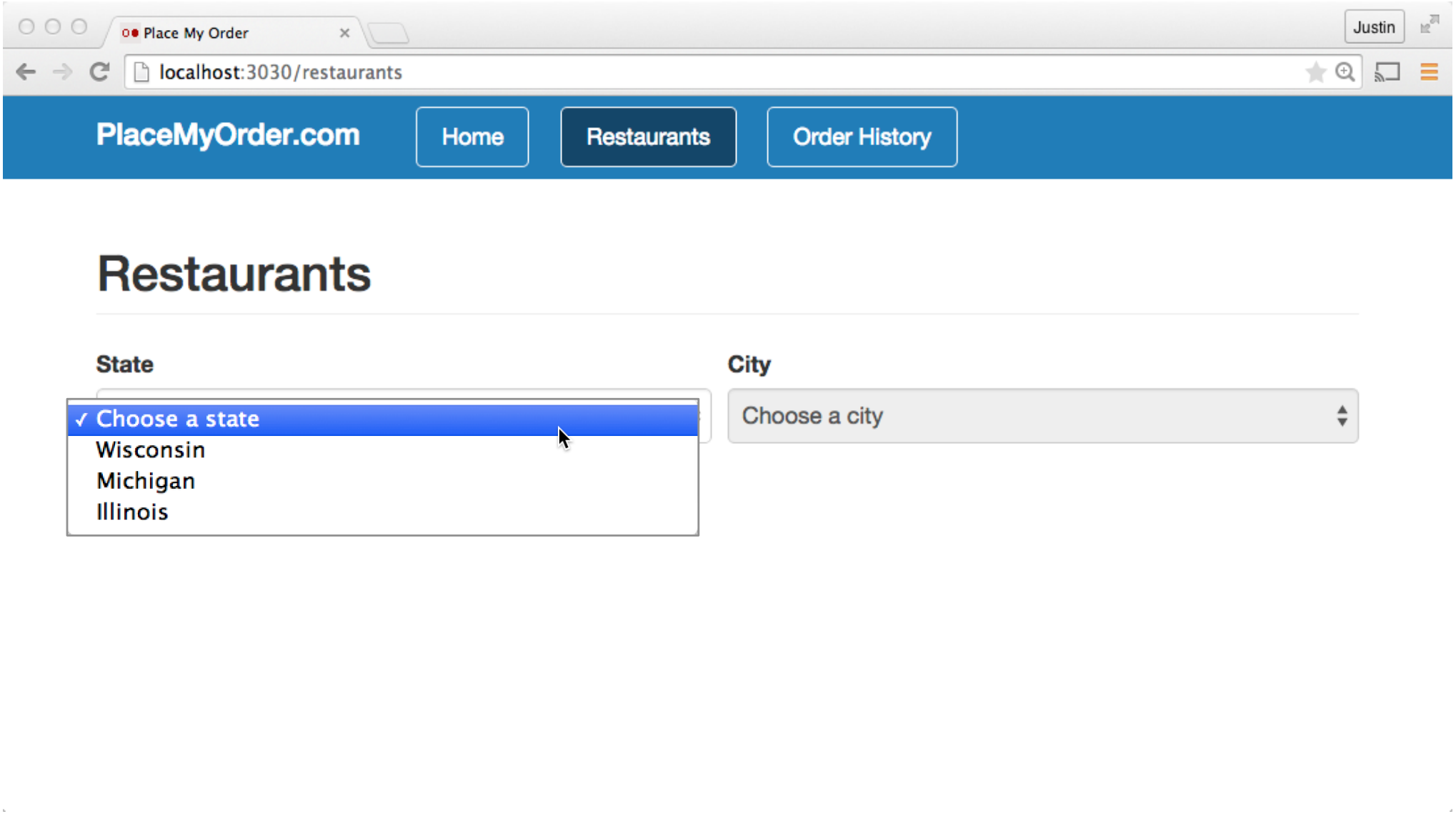
```
<can-component tag="hello-world">
    <style type="less">
        color: red;
    </style>
    <template> {{message}} World! </template>
    <view-model>
        export default {
            message: "Hello"
        };
    </view-model>
</can-component>
```

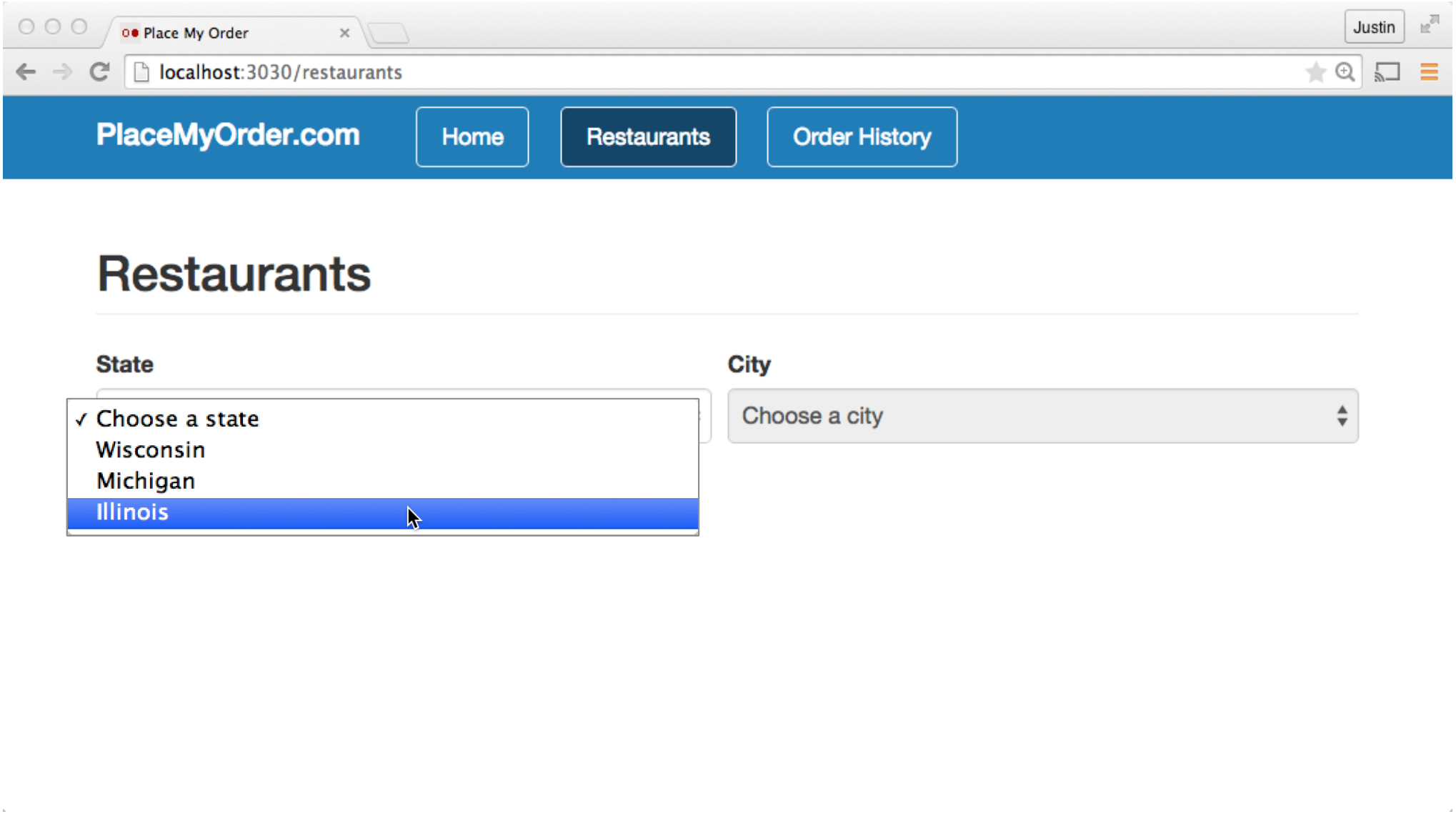
5. Unit Testable ViewModels

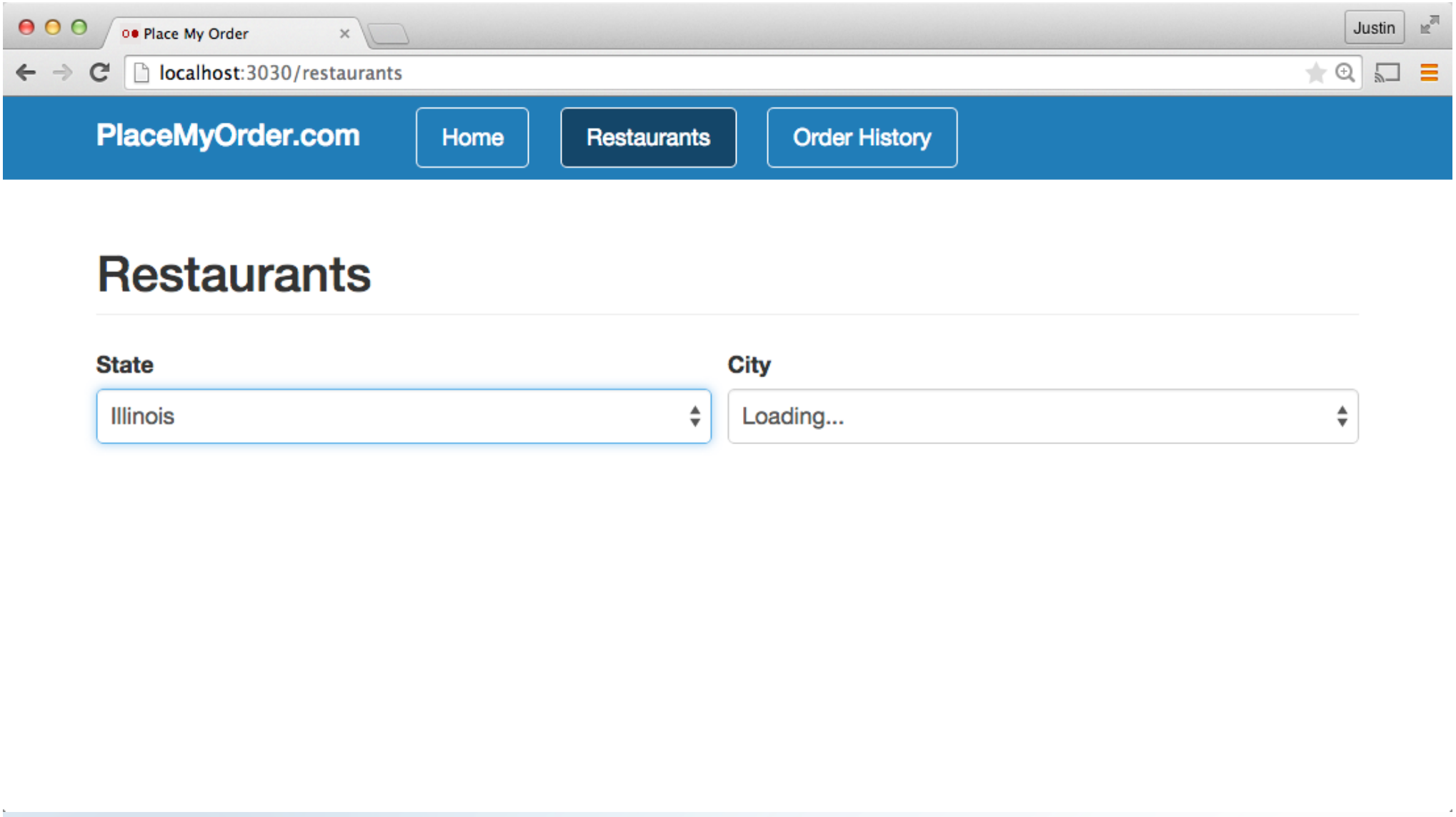
- Identify values that represent the state
- Write a test for the view model
- Write the view model
- Write the template

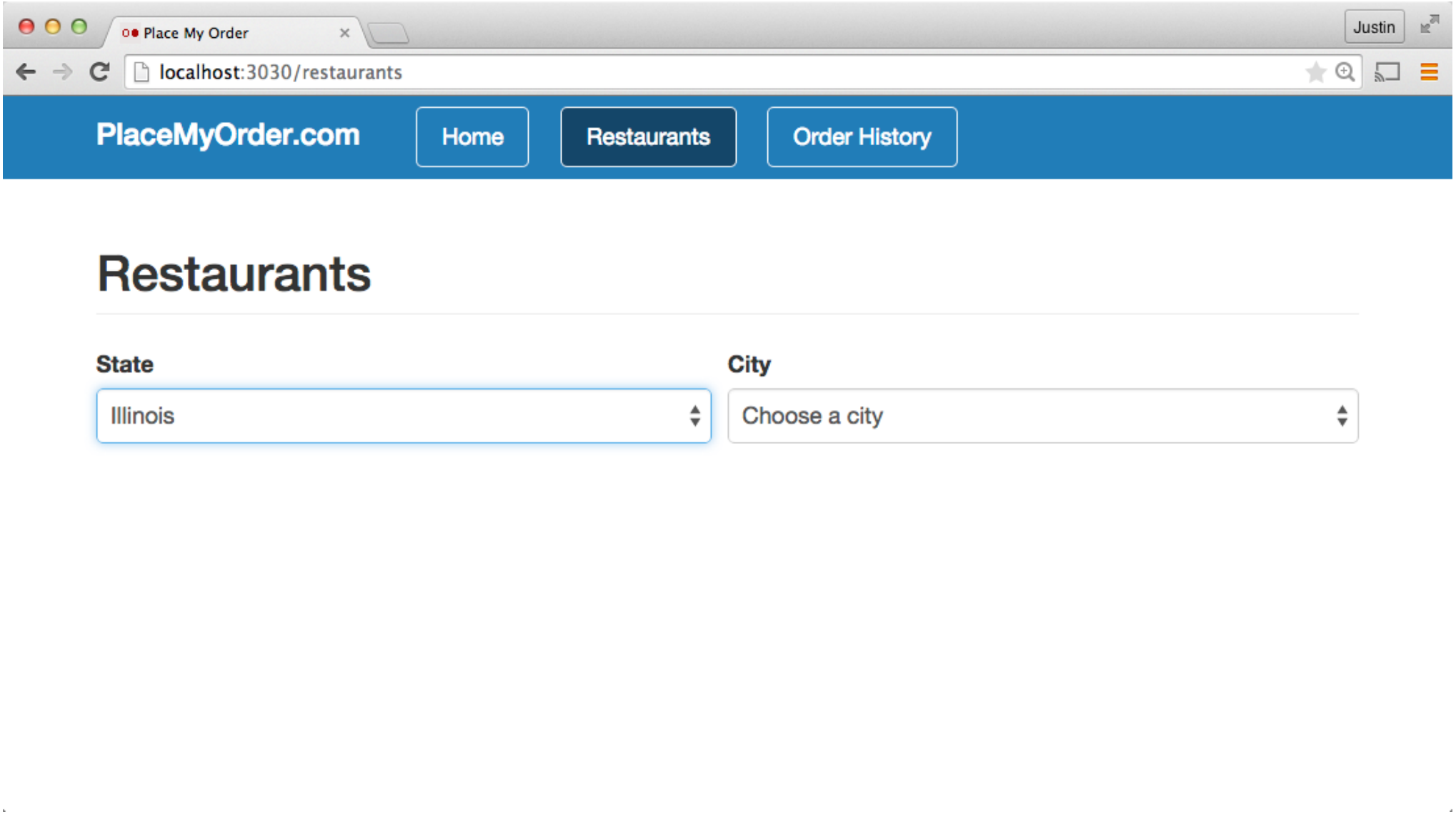

Place My Order
Justin
localhost:3030/restaurants
PlaceMyOrder.com
Home
Restaurants
Order History
Restaurants
State
Loading...
City
Choose a city

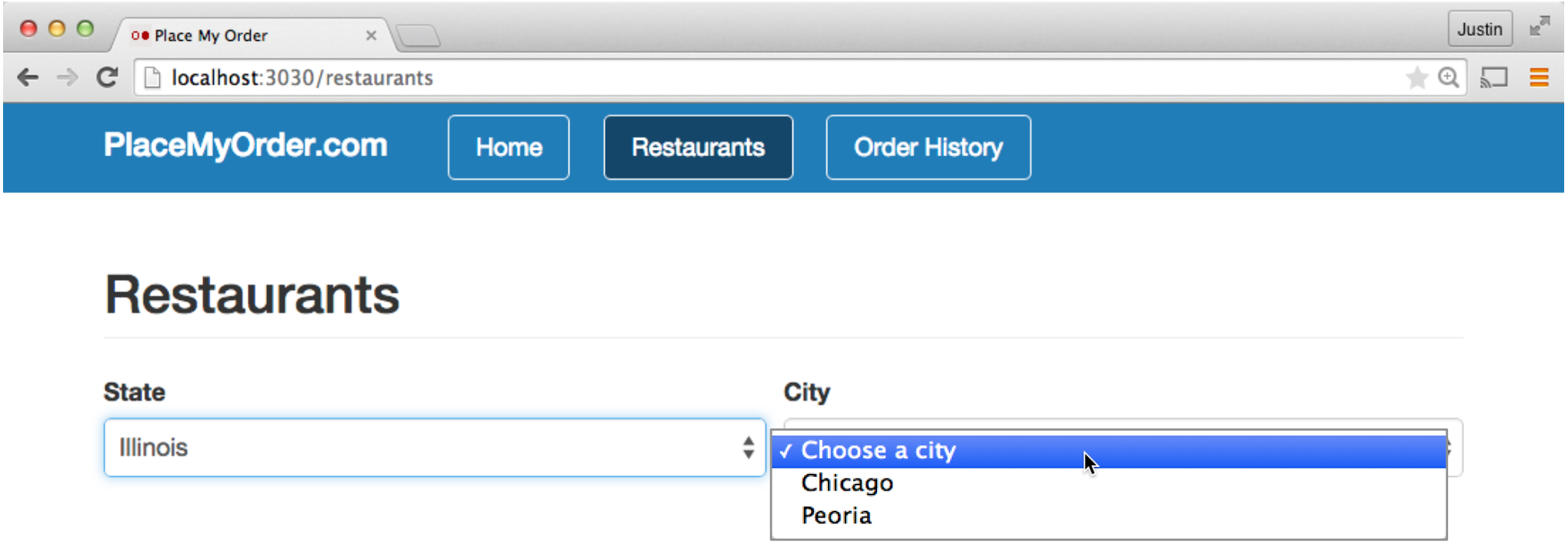

Place My Order
Justin
localhost:3030/restaurants
PlaceMyOrder.com
Home
Restaurants
Order History
Restaurants
State
Choose a state
City
Choose a city

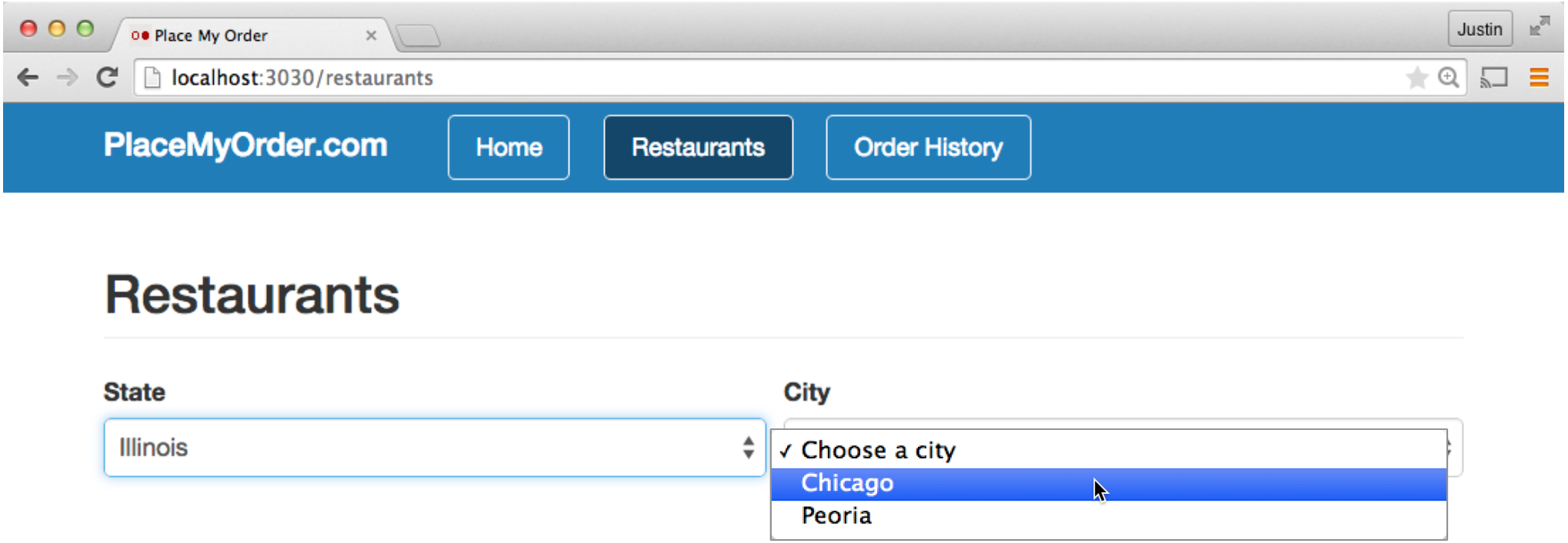

Place My Order
Justin
localhost:3030/restaurants
PlaceMyOrder.com
Home
Restaurants
Order History
Restaurants
State
City
Choose a state
Wisconsin
Michigan
Illinois
Choose a city


Place My Order
Justin
localhost:3030/restaurants
PlaceMyOrder.com
Home
Restaurants
Order History
Restaurants
State
City
Choose a state
Wisconsin
Michigan
Illinois
Choose a city


Place My Order
Justin
localhost:3030/restaurants
PlaceMyOrder.com
Home
Restaurants
Order History
Restaurants
State
Illinois
City
Loading...


Place My Order
Justin
localhost:3030/restaurants
PlaceMyOrder.com
Home
Restaurants
Order History
Restaurants
State
Illinois
City
Choose a city


Place My Order
Justin
localhost:3030/restaurants
PlaceMyOrder.com
Home
Restaurants
Order History
Restaurants
State
Illinois
City
✓ Choose a city
Chicago
Peoria


Place My Order
Justin
localhost:3030/restaurants
PlaceMyOrder.com
Home
Restaurants
Order History
Restaurants
State
Illinois
City
✓ Choose a city
Chicago
Peoria

Place My Order

Justin

localhost:3030/restaurants

PlaceMyOrder.com

Home | Restaurants | Order History

Restaurants

State

Illinois

City

Chicago

Cow Barn

285 W Adams Ave
Chicago, IL 60045

$$$
Hours: M-F 10am-11pm
Open Now

Place My Order

Bagel Barn

1601-1625 N Campbell Ave

$$$

Place My Order

Justin

localhost:3030/restaurants

PlaceMyOrder.com

Home

Restaurants

Order History

Restaurants

Wisconsin
Michigan
✓ Illinois

City

Chicago

Cow Barn

285 W Adams Ave
Chicago, IL 60045

$$$
Hours: M-F 10am-11pm
Open Now

Place My Order

Bagel Barn

Place My Order

Justin

localhost:3030/restaurants

PlaceMyOrder.com

Home

Restaurants

Order History

Restaurants

Wisconsin
Michigan
✓ Illinois

City

Chicago

Cow Barn

285 W Adams Ave
Chicago, IL 60045

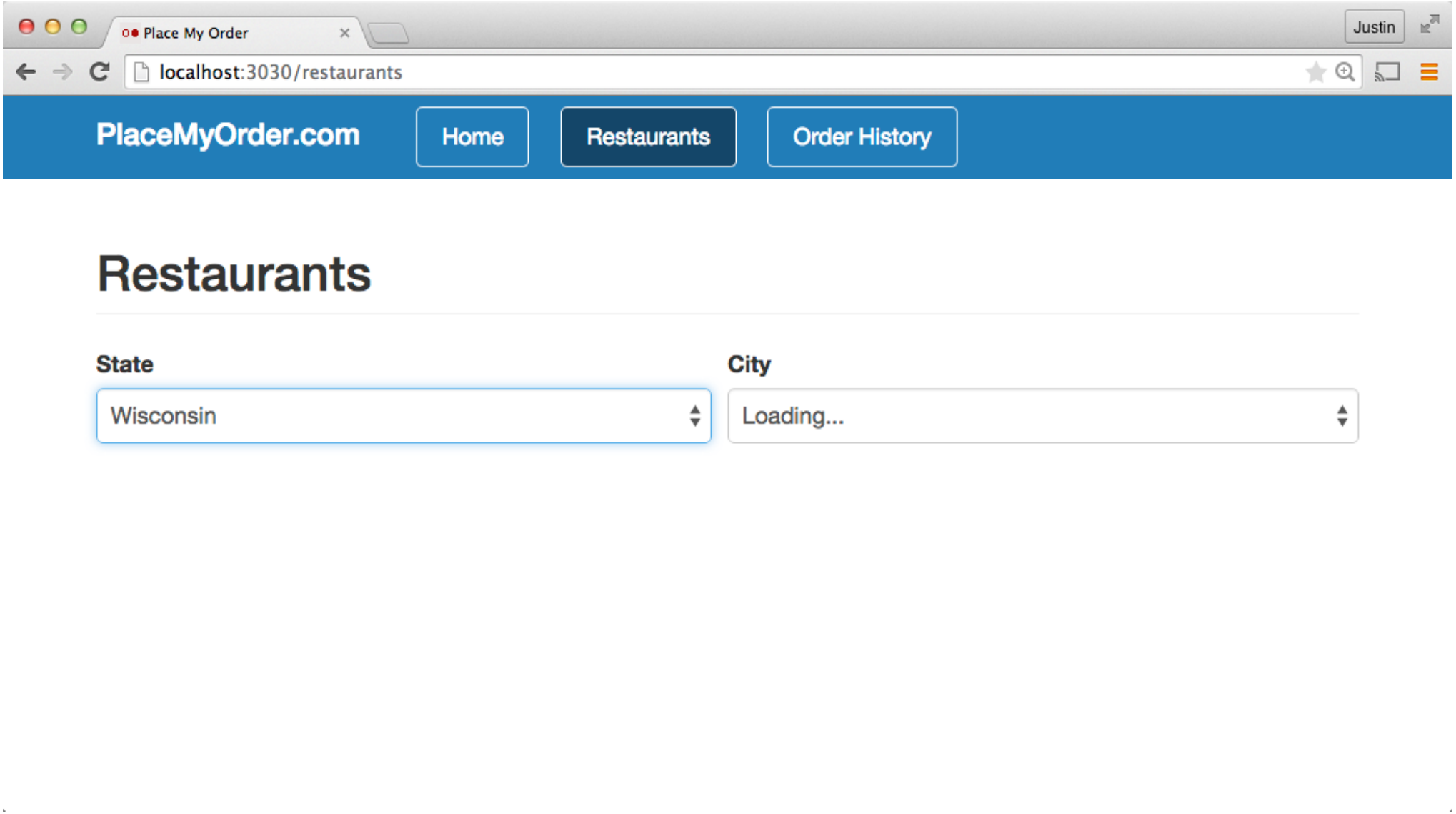
$$$
Hours: M-F 10am-11pm
Open Now

Place My Order

Bagel Barn

1601-1625 N Campbell Ave

$$$


Place My Order
Justin
localhost:3030/restaurants
PlaceMyOrder.com
Home
Restaurants
Order History
Restaurants
State
Wisconsin
City
Loading...

Place My Order

Justin

localhost:3030/restaurants

PlaceMyOrder.com

Home | Restaurants | Order History

Restaurants

State

Wisconsin

City

Choose a city

Testable View Models

1. Identify Values That Represent The State

PlaceMyOrder.com

Home | Restaurants | Order History

Restaurants

State

Loading...

City

Choose a city

```
{
 states: Promise



}
```

PlaceMyOrder.com

Home | Restaurants | Order History

Restaurants

State

✓ Choose a state
Wisconsin
Michigan
Illinois

City

Choose a city

```
{
 states: Promise   <[State]>


}
```

PlaceMyOrder.com | Home | Restaurants | Order History

Restaurants

State

Illinois

City

Loading...

```
{
 states: Promise<[State]>
 state: String "IL",
 cities: Promise<[City]>,


}
```

PlaceMyOrder.com | Home | Restaurants | Order History

Restaurants

State

Illinois

City

✓ Choose a city
Chicago
Peoria

```
{
 states: Promise<[State]>
 state: String "IL",
 cities: Promise<[City]>,
 city: String "Chicago",

}
```

PlaceMyOrder.com

Home | Restaurants | Order History

Restaurants

State

Illinois

City

Chicago

Cow Barn

285 W Adams Ave
Chicago, IL 60045

$$$
Hours: M-F 10am-11pm
Open Now

Place My Order

```
{
 states: Promise<[State]>
 state: String "IL",
 cities: Promise<[City]>,
 city: String "Chicago",
 restaurants: Promise<[Restaurant]>
}
```

Testable ViewModels

1. Identify values that represent the state
2. Test the ViewModel

place_my_order/restaurant/list_test.js

```
import {ViewModel as RestaurantListVM} from "./list";
import QUnit from 'steal-qunit';
import fixture from 'can/util/fixture/';

QUnit.module("place-my-order/restaurant/list");

QUnit.asyncTest("basics", function(){

    ...

});
```

place_my_order/restaurant/list_test.js

```
QUnit.asyncTest("basics", function(){

    var states = [{ name: "Calisota", short: "CA" },
                  { name: "New Troy", short: "NT"}];
    var caCities = [{state: "CA",name: "Casadina"}];
    var ntCities = [{state: "NT", name: "Alberny"}];
    var casadinaRestaurants = [{
   "_id":"1","name":
   "Cheese City","slug":"cheese-city”
  }];

  …

});
```

place_my_order/restaurant/list_test.js

```
QUnit.asyncTest("basics", function(){
      …
      var casadinaRestaurants = [{ … }];

      fixture({
        "/api/states": ()=> ({data: states}),
        "/api/cities": function(request){
            return request.data.state === "CA" ? caCities : ntCities;
     },
        "/api/restaurants": ()=> ({data: casadinaRestaurants})
      })

    …

});
```

place_my_order/restaurant/list_test.js

```
QUnit.asyncTest("basics", function(){
    …
    var casadinaRestaurants = [{ … }];
    fixture({…})

  var rlVM = new RestaurantListVM();

    rlVM.attr("states").then(function(vmStates){
        QUnit.deepEqual(vmStates.attr(), states, "Got states");

    });


});
```

place_my_order/restaurant/list_test.js

```
QUnit.asyncTest("basics", function(){
    …
    rlVM.attr("states").then(function(vmStates){
        QUnit.deepEqual(vmStates.attr(), states, "Got states");
        rlVM.attr("state","CA");
    });

  rlVM.one("cities", function( ev, citiesPromise ) {
        citiesPromise.then(function( vmCities ) {
            deepEqual(vmCities.attr(), caCities, "Got ca cities");
            rlVM.attr("city", "Casadina");
        });
    });

});
```

place_my_order/restaurant/list_test.js

```
QUnit.asyncTest("basics", function(){
    …
    rlVM.one("restaurants", function(ev, restaurantsPromise){
        restaurantsPromise.then(function(vmRestaurants){
            deepEqual(vmRestaurants.attr(), casadinaRestaurants);

            rlVM.attr("state","NT");
            ok( !rlVM.attr("city"), "city selection removed" );
            rlVM.attr("cities").then(function(vmCities){
                deepEqual(vmCities.attr(), ntCities);
                start();
            });
        });
    });
});
```

Testable ViewModels

1. Identify values that represent the state
2. Test the ViewModel
3. Write the ViewModel

place_my_order/restaurant/list.js

```
import Map from 'can/map/';

export const ViewModel = Map.extend({
  define: {
    states: {
      value() {
        return State.findAll({});
      }
    }




  }
});
```

place_my_order/restaurant/list.js

```
define: {
  states: { … },
  state: {
    set() {
      // Remove the city when the state changes
      this.removeAttr('city');
    }
  },
  cities: {
    get() {
      var state = this.attr('state');
      return state ? City.findAll({ state }) : null;
    }
  }
}
```

place_my_order/restaurant/list.js

```
define: {
  states: { … },
  state: { … },
  cities: { … } ,
  city: { value: null },
  restaurants: {
    get: function(){
      var state = this.attr('state'),
          city = this.attr('city');

      return state && city ?
             Restaurant.findAll({ state, city }) : null;
    }
  }
}
```

Testable ViewModels

1. Identify values that represent the state
2. Test the ViewModel
3. Write the ViewModel
4. Write the template

place_my_order/restaurant/list.stache

```
<label>State</label>
<select can-value="{state}" {{#if states.isPending}}disabled{{/if}}>
  {{#if states.isPending}}
    <option>Loading...</option>
  {{else}}
    {{^state}}
      <option>Choose a state</option>
    {{/state}}
    {{#each states.value}}
      <option value="{{short}}">{{name}}</option>
    {{/each}}
  {{/if}}
</select>
```

Benefits

- Unit tests without a DOM.
- Makes you think about describing the behavior as a set of data changes.

6. Two way routing

- Differences from one way routing
- Example use
- Benefits

```
/restaurants/cheese-city
```

```
route(':page');
route(':page/:slug');
route(':page/:slug/:action');
```

```
{
        page: 'restaurants',
        slug: 'cheese-city'
}
```

```
/restaurants/cheese-city/order
```

```
route(':page');
route(':page/:slug');
route(':page/:slug/:action');
```

```
{
        page: 'restaurants',
        slug: 'cheese-city',
        action: 'order'
}
```

```
{
        page: 'order-history'
}
```

```
route(':page');
route(':page/:slug');
route(':page/:slug/:action');
```

/order-history

```
{
        page: 'restaurants',
        slug: 'burger-bar',
        action: 'confirm'
}
```

```
route(':page');
route(':page/:slug');
route(':page/:slug/:action');
```

```
/restaurants/burger-bar/confirm
```

```
{
        page: 'restaurants',
        state: 'IL',
        city: 'Chicago'
}
```

```
route(':page');
route(':page/:slug');
route(':page/:slug/:action');
```

```
/restaurants?state=IL&city=Chicago
```

main.stache

```
{{#routeMatches "/home"}}
  <app-home></app-home>
{{/}}
{{#routeMatches "/restaurants"}}
  <app-restaurant-list></app-restaurant-list>
{{/}}
```

main.stache

```
{{#eq route.page "home"}}
   <app-home></app-home>
{{/eq}}
{{#eq route.page "restaurants"}}
  <app-restaurant-list></app-restaurant-list>
{{/eq}}
```

place_my_order/restaurant/list.stache

```
{{#each restaurants.value}}
  <div class="restaurant">
    <h3>{{name}}</h3>
    <a href="/restaurants/{{slug}}">
      Place My Order
    </a>
  </div>
{{/each}}
```

```
href="/restaurants/cheese-city"
```

place_my_order/restaurant/list.stache

```
{{#each restaurants.value}}
  <div class="restaurant">
    <h3>{{name}}</h3>
    <a can-href="{ page='restaurants' slug=slug }">
      Place My Order
    </a>
  </div>
{{/each}}
```

```
href="/restaurants/cheese-city"
```

place_my_order/routes.js

```
route(':page');
route(':page/:slug');
route(':page/:slug/:action');
```

place_my_order/routes.js

```
route(':page');
route(':page/restaurant/:slug');
route(':page/restaurant/:slug/:action');
```

place_my_order/restaurant/list.stache

```
{{#each restaurants.value}}
  <div class="restaurant">
    <h3>{{name}}</h3>
    <a can-href="{ page='restaurants' slug=slug }">
      Place My Order
    </a>
  </div>
{{/each}}
```

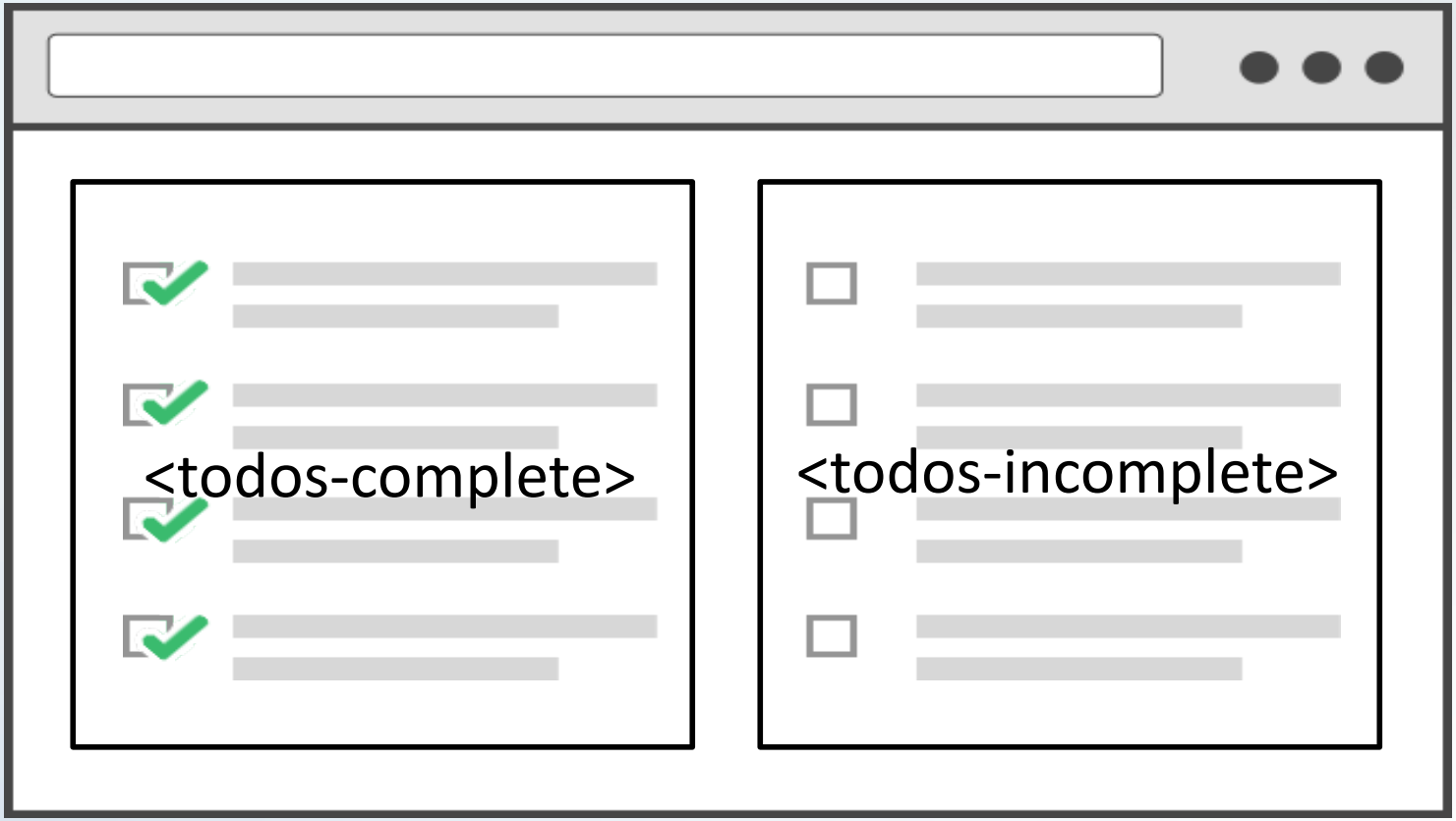
```
href="/restaurants/restaurant/cheese-city"
```

Benefits

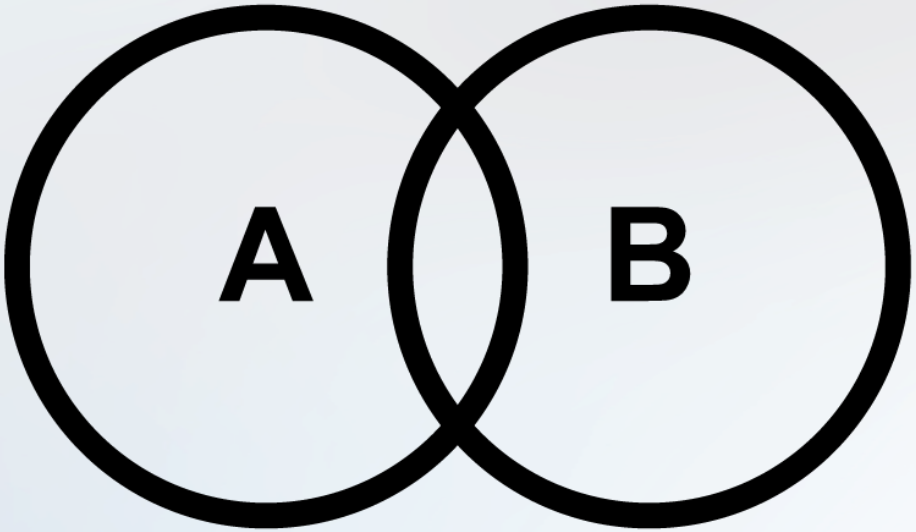
- Keep routing rules separate from application logic.
- Update routing rules independent of the page.
- Makes you think about describing the route’s stateful properties.

7. Super Models

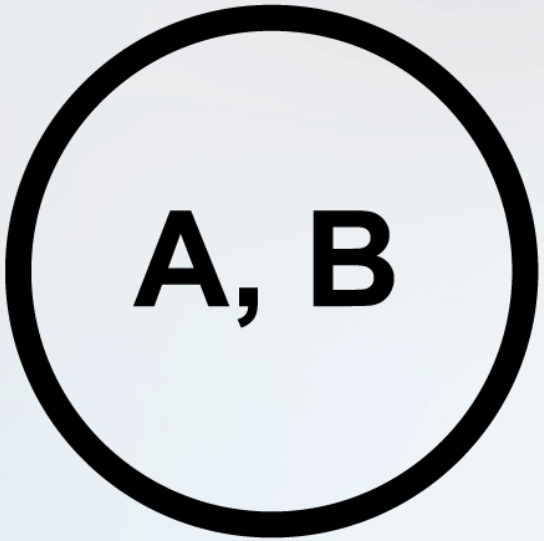
The Problem



$A = B$	Equal
$A \subseteq B$	Subset
$A \subset B$	Proper Subset
$A \setminus B$	Difference
$A \cap B$	Intersection
$A \cup B$	Union

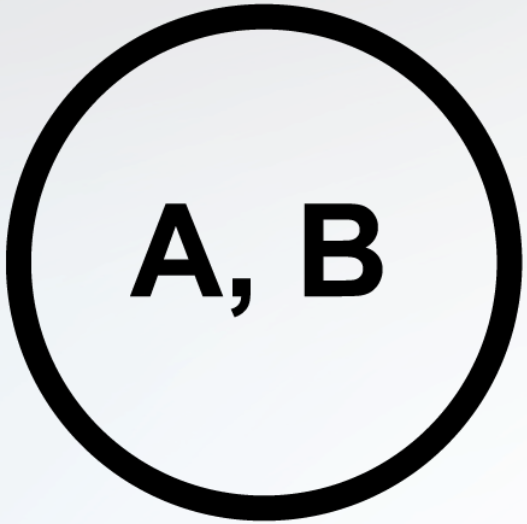


A = B	Equal
$A \subseteq B$	Subset
$A \subset B$	Proper Subset
A \ B	Difference
$A \cap B$	Intersection
$A \cup B$	Union



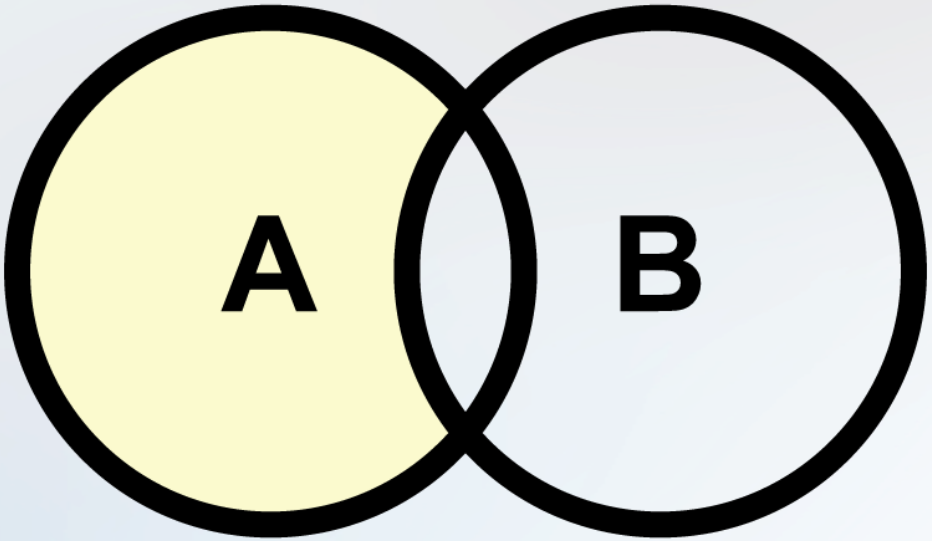
A = B	Equal
$A \subseteq B$	Subset
$A \subset B$	Proper Subset
A \ B	Difference
$A \cap B$	Intersection
$A \cup B$	Union

or

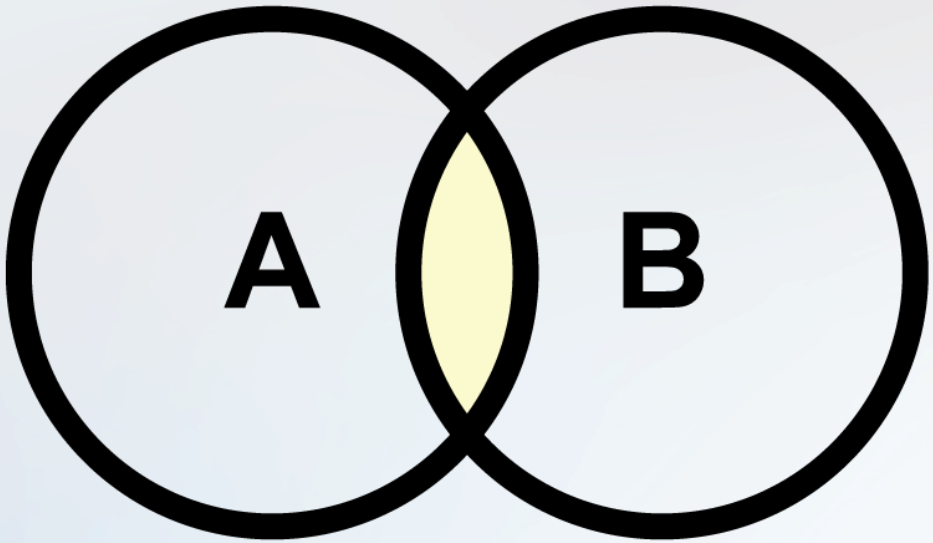


$A = B$	Equal
$A \subseteq B$	Subset
$B \subset A$	Proper Subset
$A \setminus B$	Difference
$A \cap B$	Intersection
$A \cup B$	Union

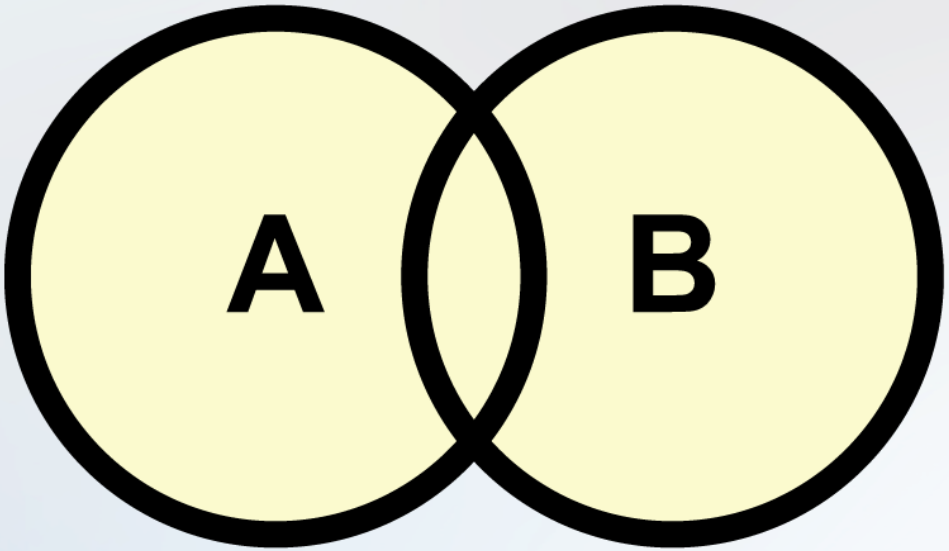
$A = B$	Equal
$A \subseteq B$	Subset
$A \subset B$	Proper Subset
$A \setminus B$	Difference
$A \cap B$	Intersection
$A \cup B$	Union



$A = B$	Equal
$A \subseteq B$	Subset
$A \subset B$	Proper Subset
$A \setminus B$	Difference
$A \cap B$	Intersection
$A \cup B$	Union



$A = B$	Equal
$A \subseteq B$	Subset
$A \subset B$	Proper Subset
$A \setminus B$	Difference
$A \cap B$	Intersection
$A \cup B$	Union

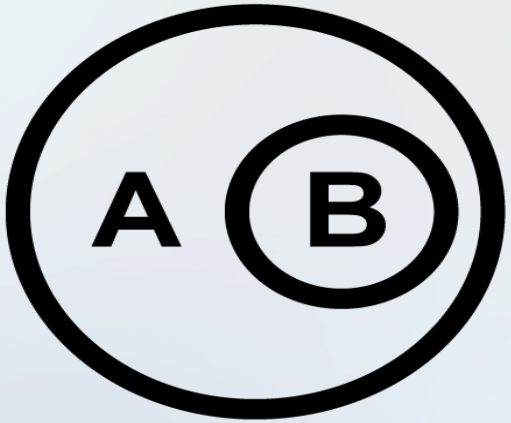


```
var setA = {
  userId: 5,
  completed: true
};

$.get("/api/todos", setA, function(todos){ ... })
//-> api/todos?userId=5&completed=true
```

```
var setA = {
  userId: 5,
  completed: false
};
```

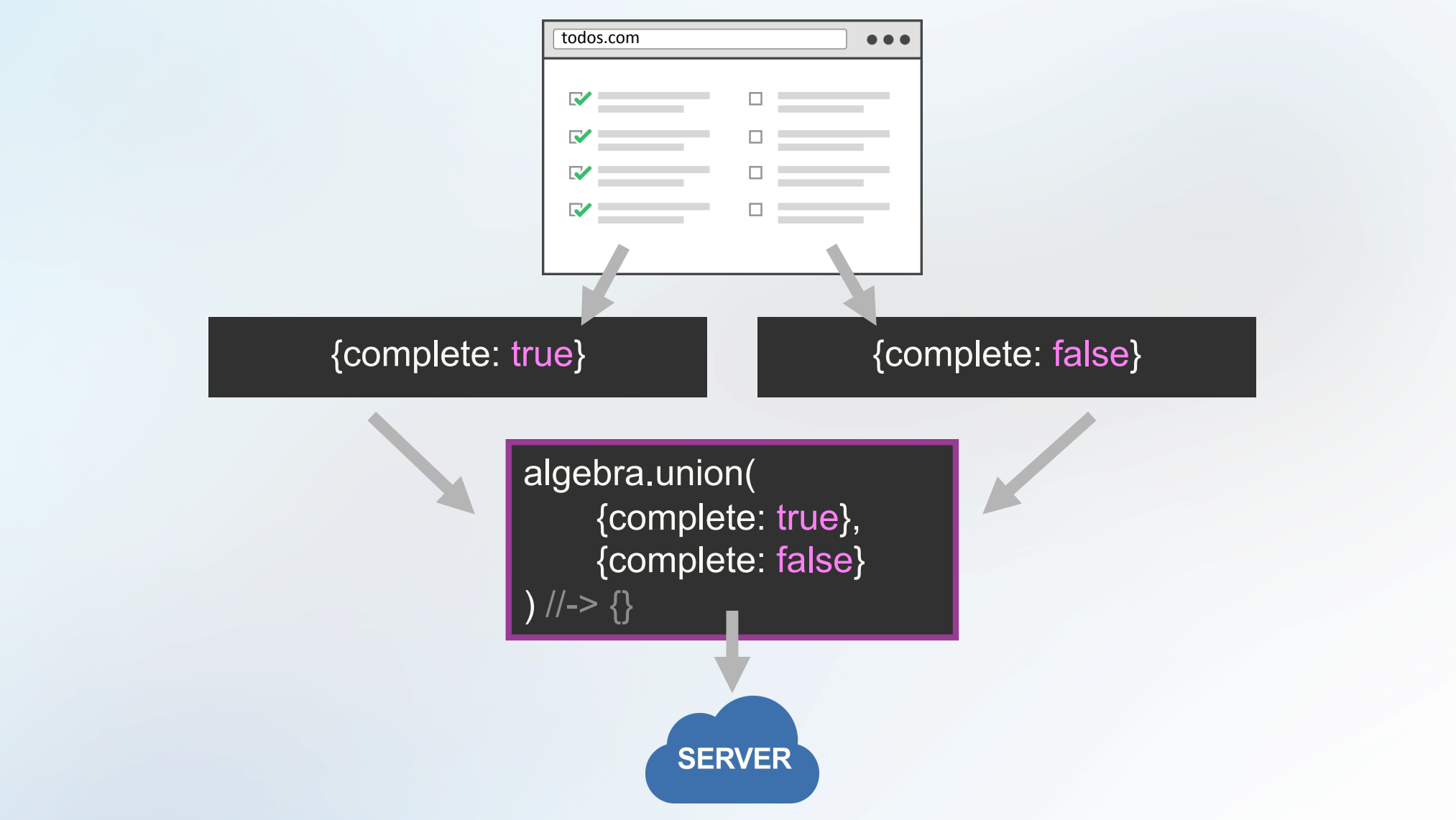
```
var setB = {
  userId: 5,
  completed: false,
  type: 'critical'
};
```

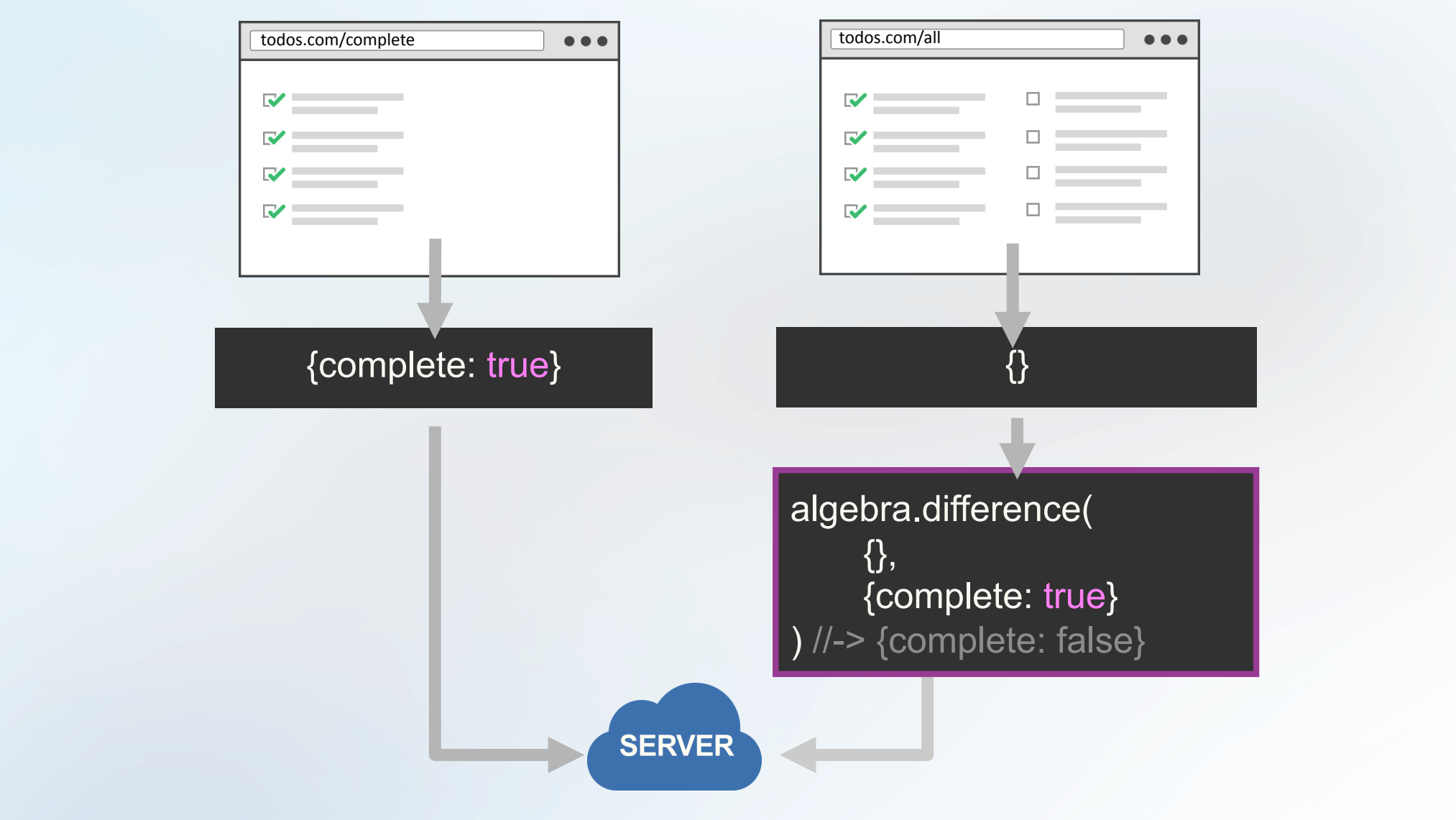


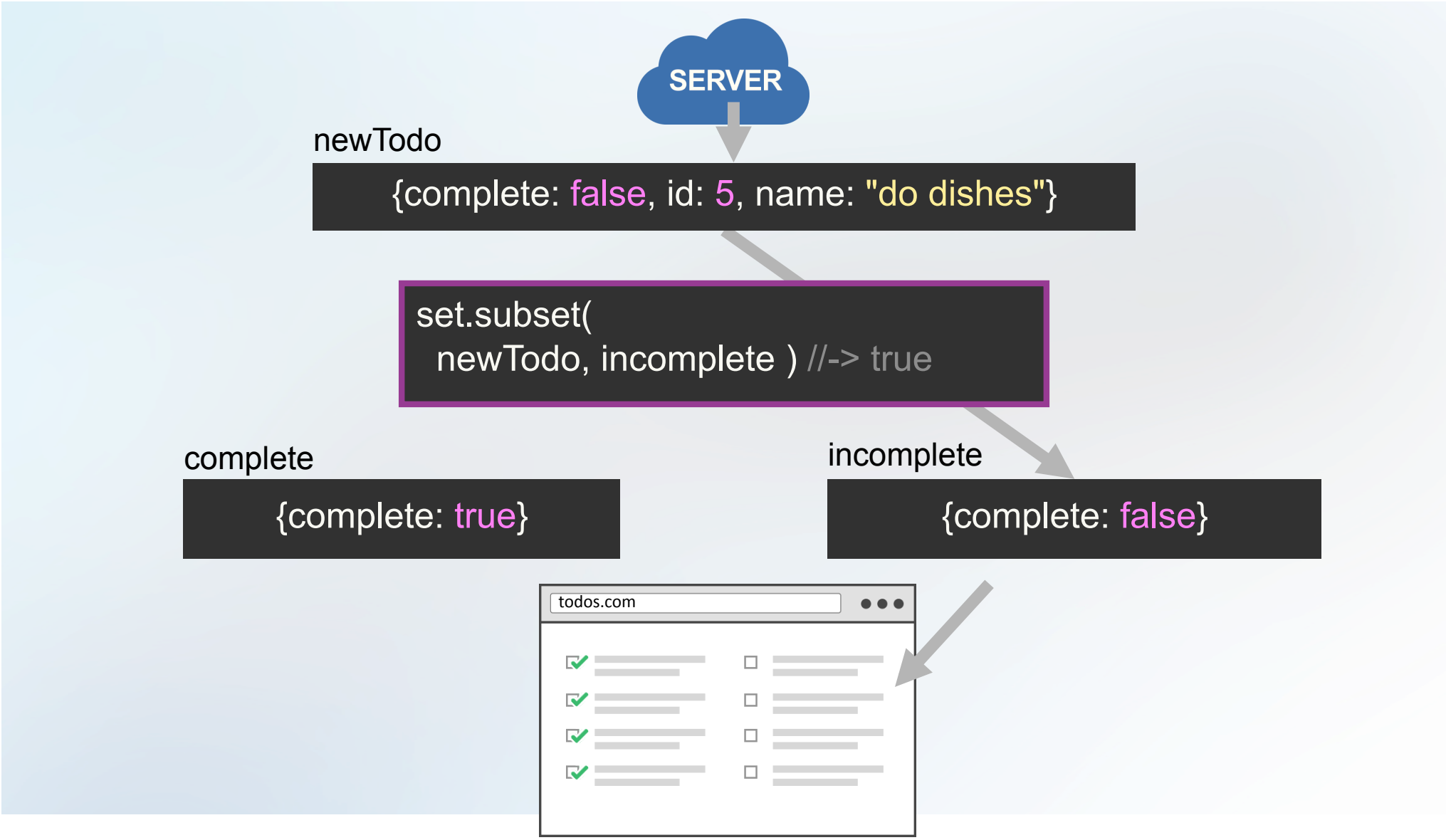

```
// B ⊆ A
set.subset(
  setB,
  setA ) //-> true
```

```
var algebra = new set.Algebra(
 set.comparators.rangeInclusive("start", "end"),
 set.comparators.enum("status", ["new","delivered"]),
 set.comparators.boolean("completed")
);

algebra.subset(
     {start: 0, end: 50},
     {start: 0, end: 100}
); //-> true
```


todos.com
{complete: true}
{complete: false}
algebra.union(
{complete: true},
{complete: false}
) //-> {}
SERVER

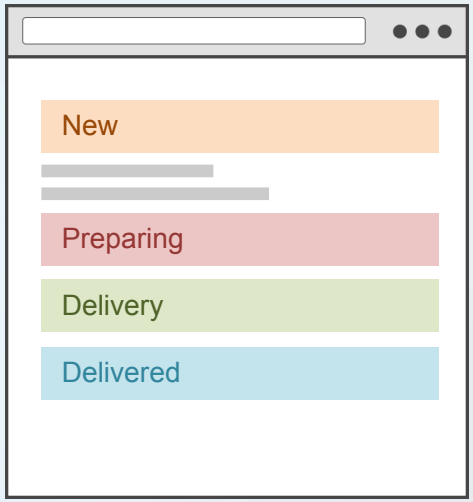

todos.com/complete
todos.com/all
{complete: true}
{}
algebra.difference(
{},
{complete: true}
) //-> {complete: false}
SERVER


SERVER
newTodo
{complete: false, id: 5, name: "do dishes"}
set.subset(
newTodo, incomplete) //-> true
complete
{complete: true}
incomplete
{complete: false}
todos.com

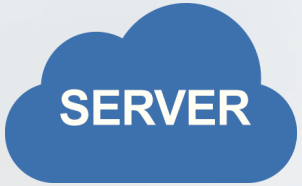
Sets give you:

- Automatic live updates
- Speed
  - inline-cache
  - combine requests
  - fall through caching

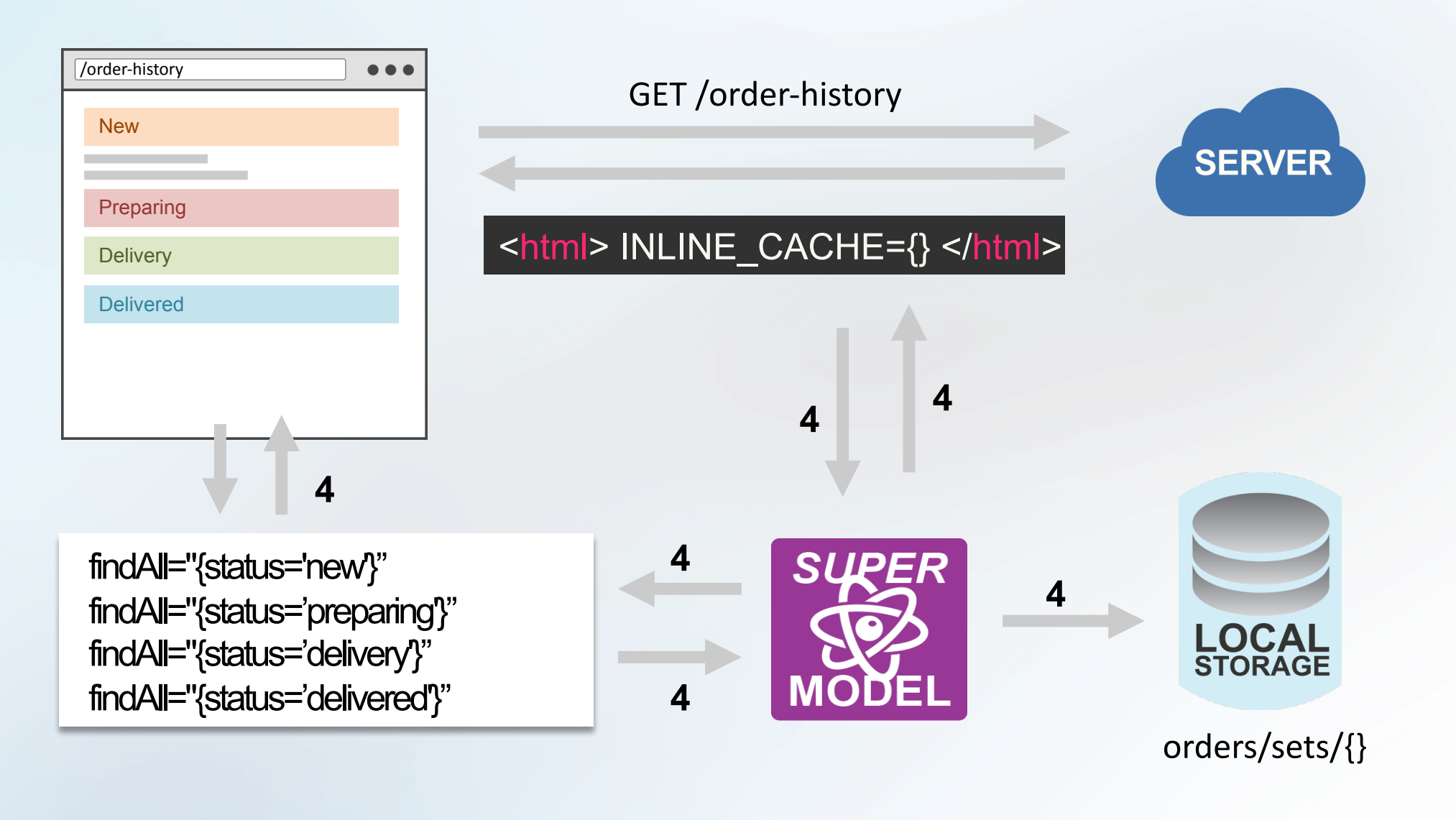
Inline Cache



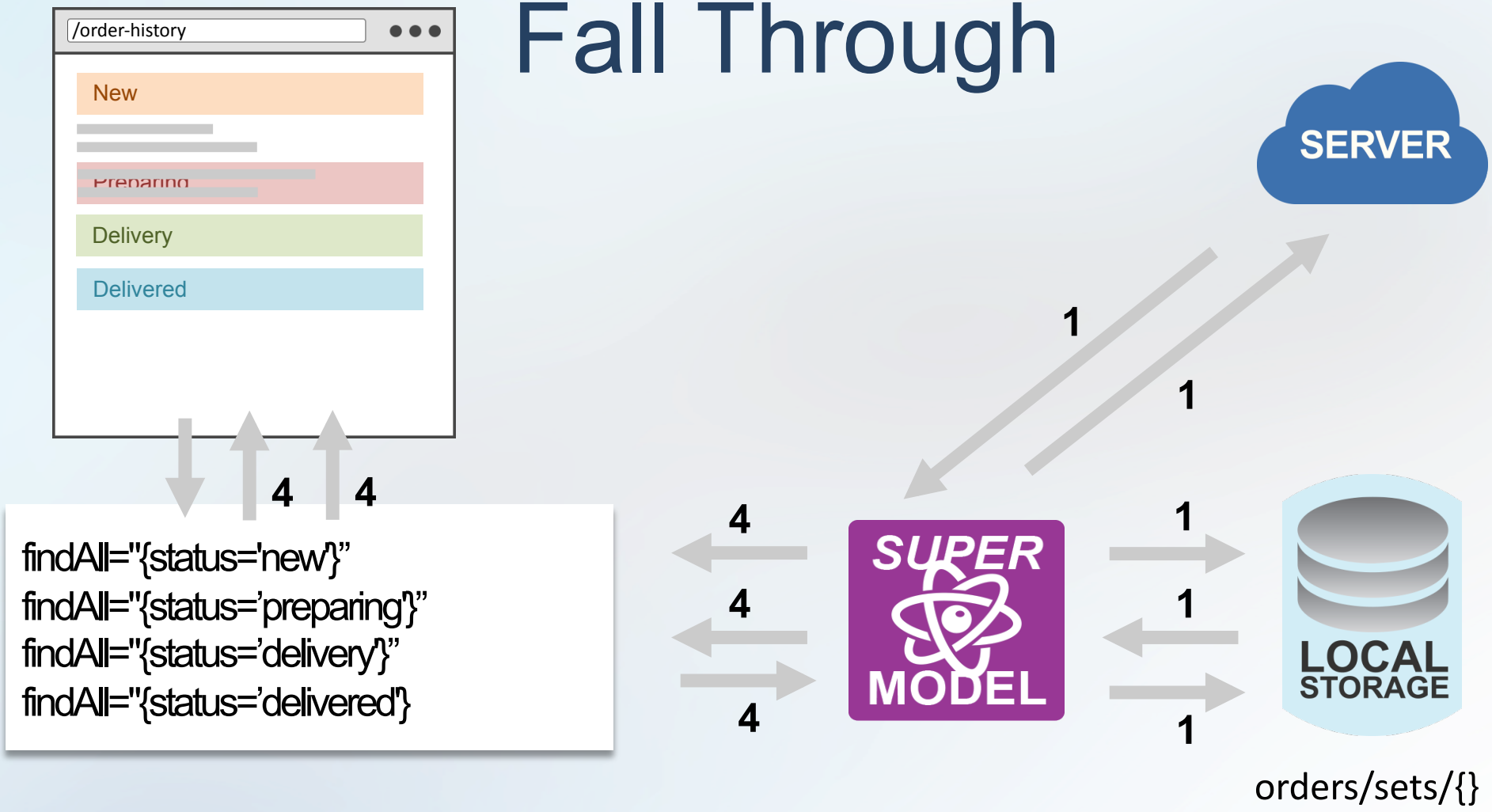
GET /order-history



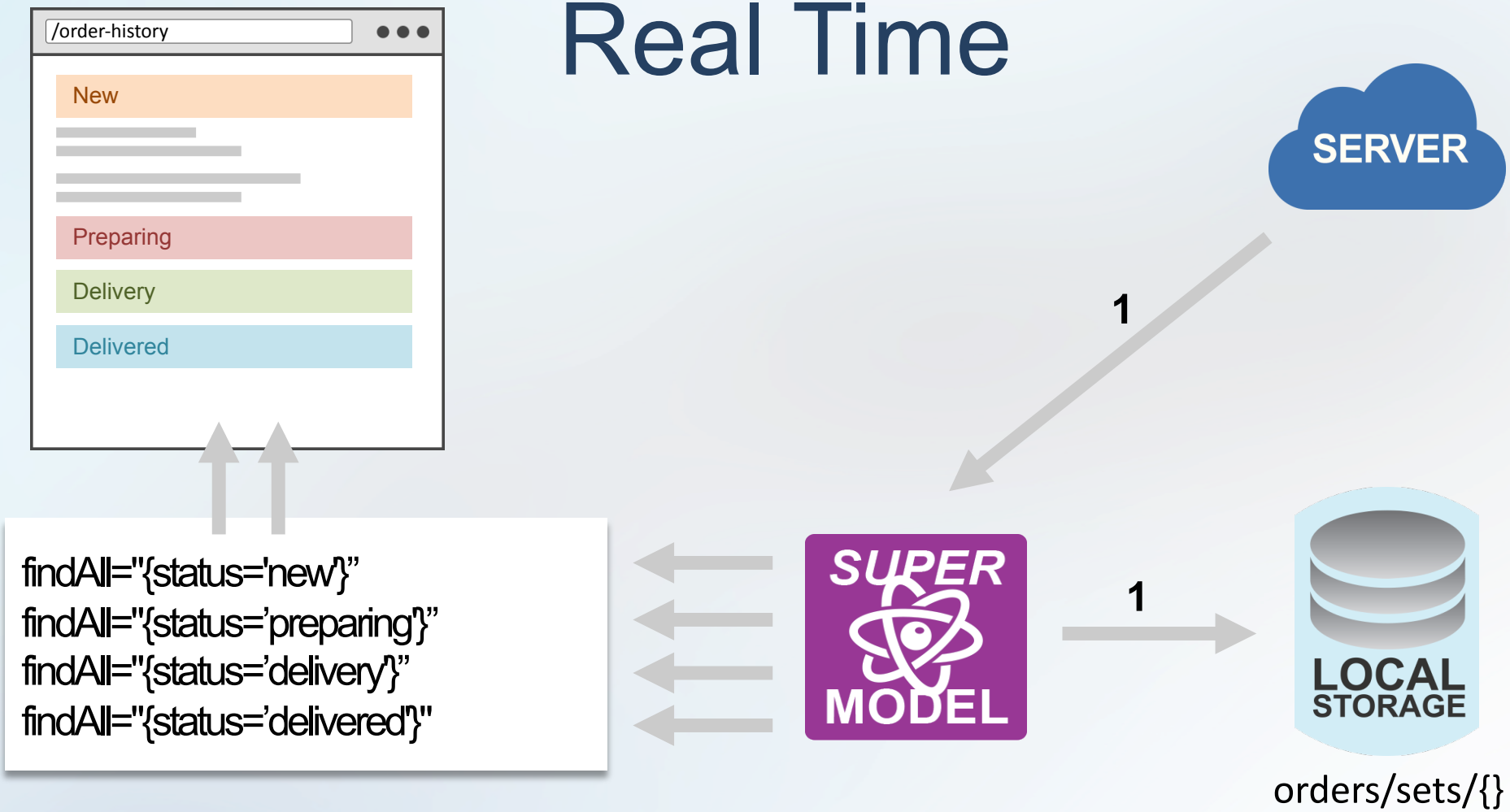

```
<html>
...
<script>
 INLINE_CACHE= {
  "orders": {
    "{status: 'new'}": [ ... ],
    "{status: 'preparing'}": [ ... ],
    ...
  }
 }
</script>
</html>
```


/order-history
New
Preparing
Delivery
Delivered
GET /order-history
SERVER
<html> INLINE_CACHE={} </html>
4
4
4
findAll="{status='new'}"
findAll="{status='preparing'}"
findAll="{status='delivery'}"
findAll="{status='delivered'}"
4
4
SUPER
MODEL
4
LOCAL
STORAGE
orders/sets/{}

Fall Through



Real Time


/order-history
New
Preparing
Delivery
Delivered
SERVER
1
findAll="{status='new'}"
findAll="{status='preparing'}"
findAll="{status='delivery'}"
findAll="{status='delivered'}"
SUPER
MODEL
1
LOCAL
STORAGE
orders/sets/{}

8. Live Reload

- Demo
- Benefits
- How it works

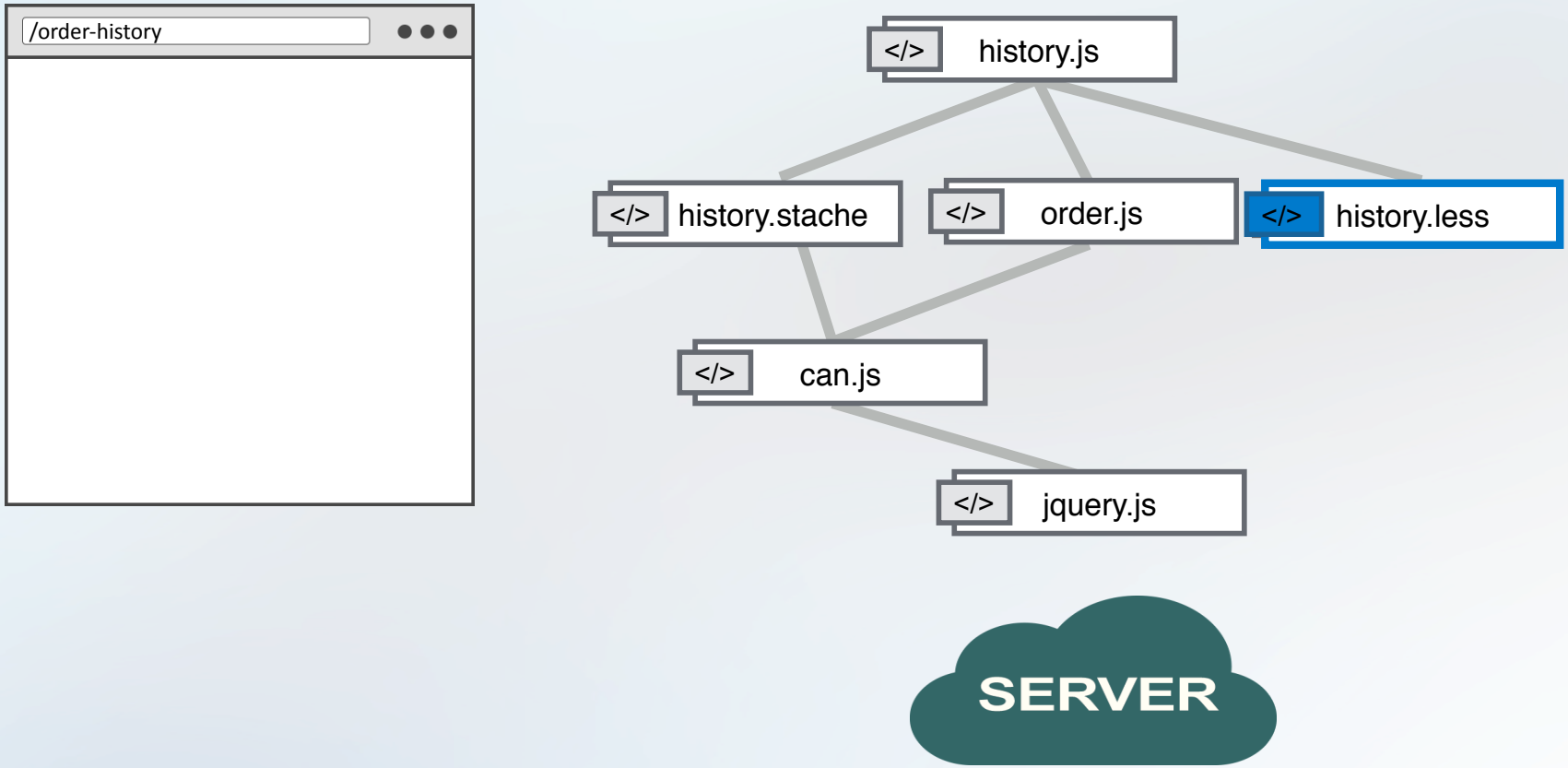
Demo

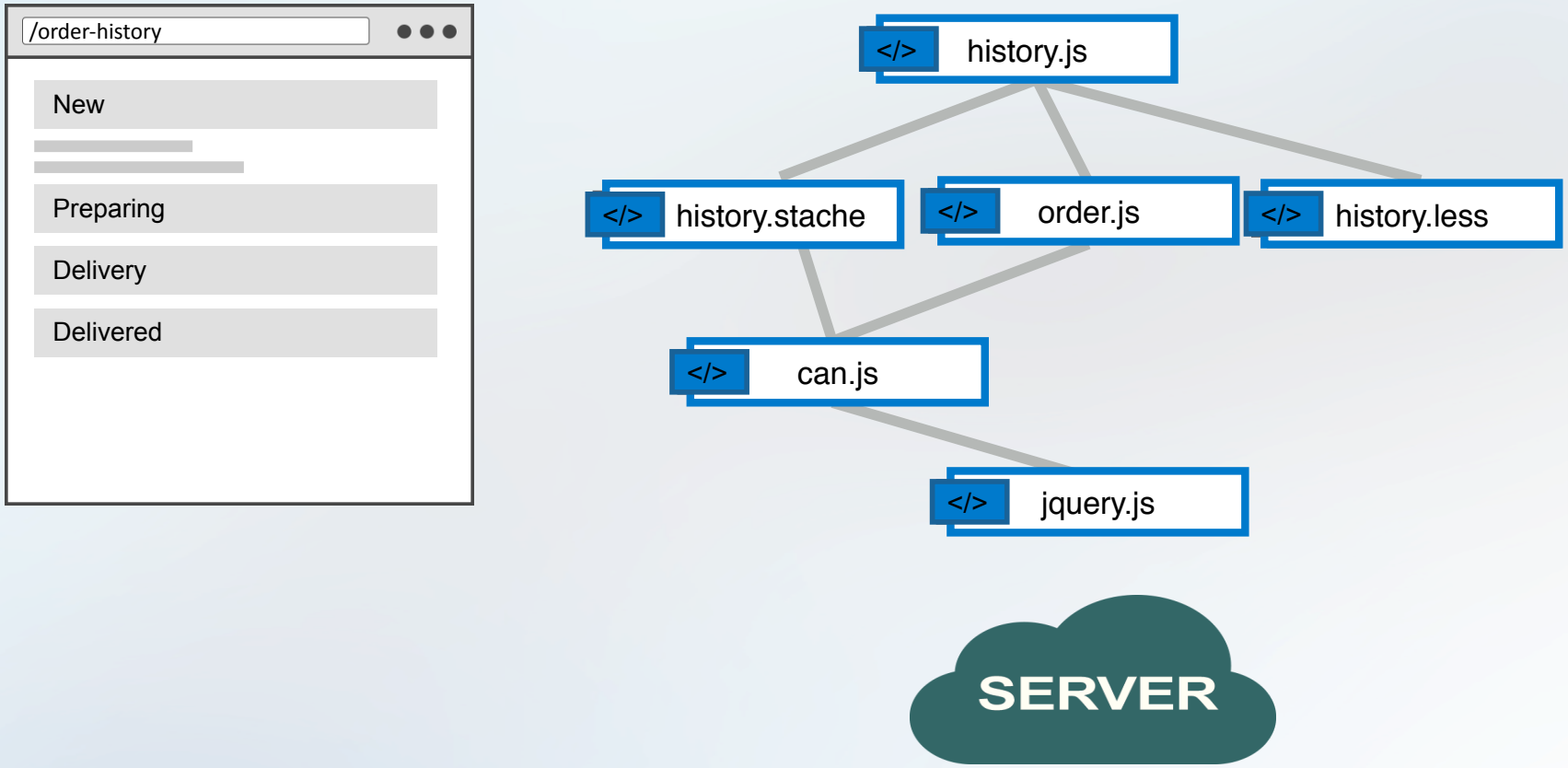
Benefits

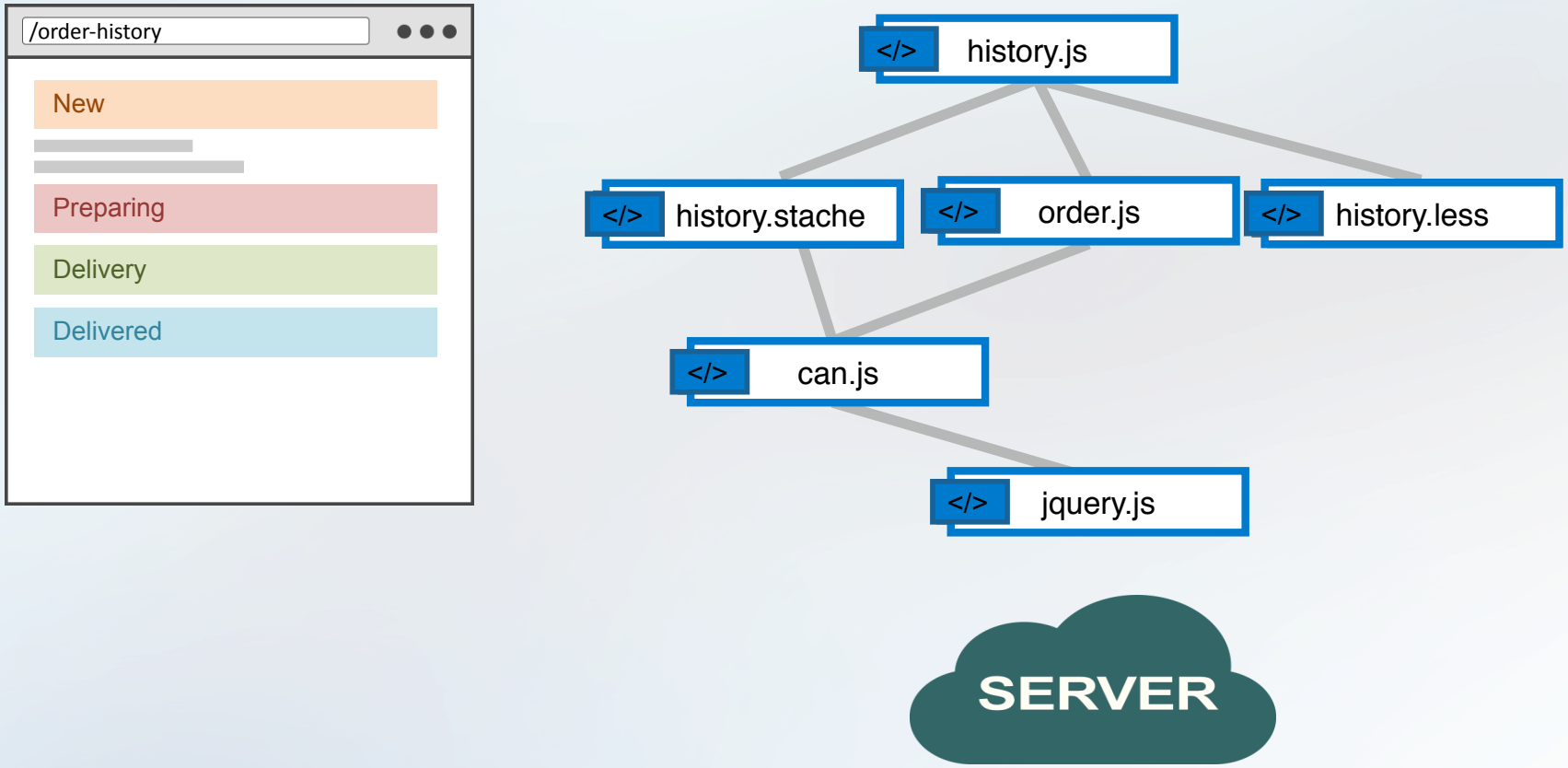
0.1 second – users feel like their actions are directly causing something to happen on the screen.

1 second - users notice the short delay, but they **stay focused on their current train of thought**.

http://www.nngroup.com/articles/powers-of-10-time-scales-in-ux/


/order-history
history.js
history.stache
order.js
history.less
can.js
jquery.js
SERVER

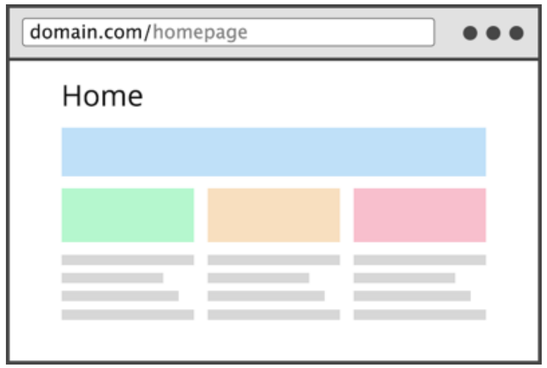

/order-history
New
Preparing
Delivery
Delivered
history.js
history.stache
order.js
history.less
can.js
jquery.js
SERVER


/order-history
New
Preparing
Delivery
Delivered
history.js
history.stache
order.js
history.less
can.js
jquery.js
SERVER

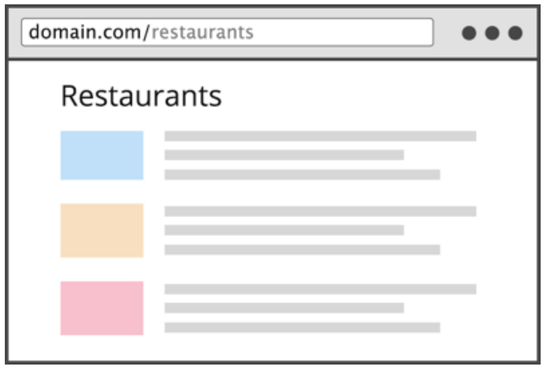
9. Smart Progressive Loading

- Difficulties loading apps
- The smart bundling algorithm
- Declaring bundle mains … in templates
- Demo (if there’s time)

Multi-Page Statically Loaded

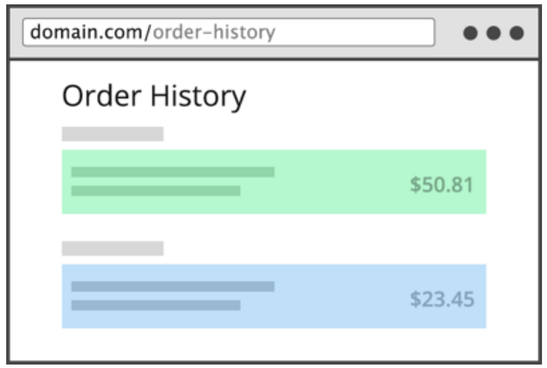


- homepage.js
  - carousel.js
  - cart.js
  - title.css
  - can.js
  - jquery.js



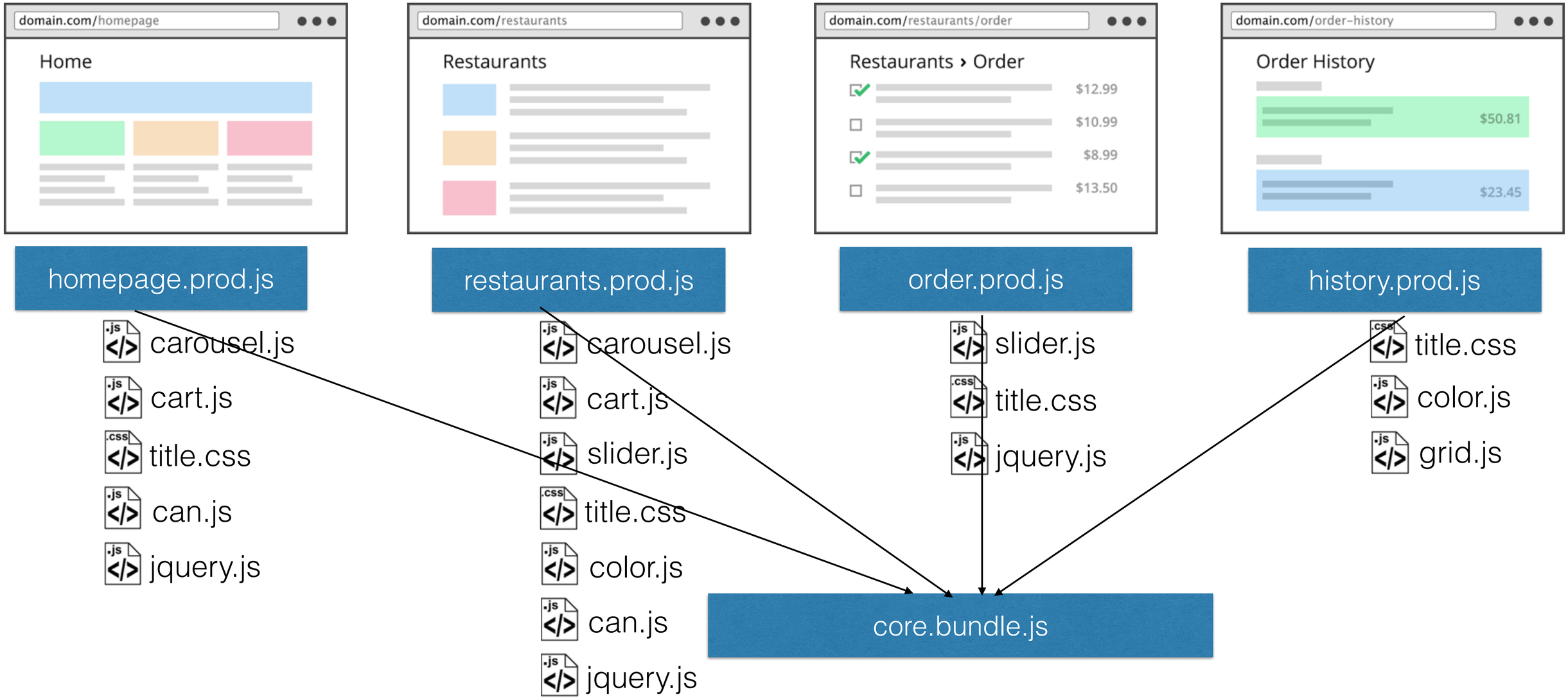
- restaurants.js
  - carousel.js
  - cart.js
  - slider.js
  - title.css
  - color.js
  - can.js
  - jquery.js

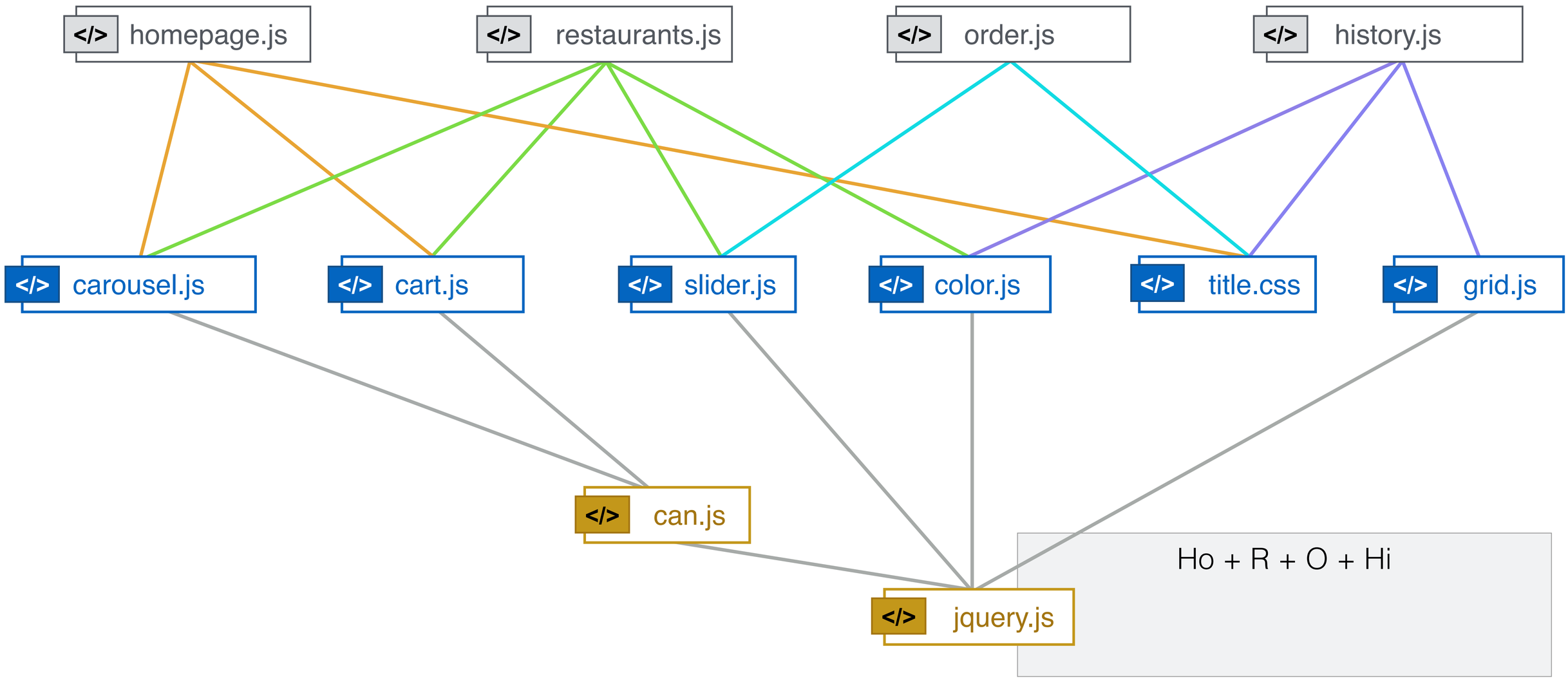
- order.js
  - slider.js
  - title.css
  - jquery.js

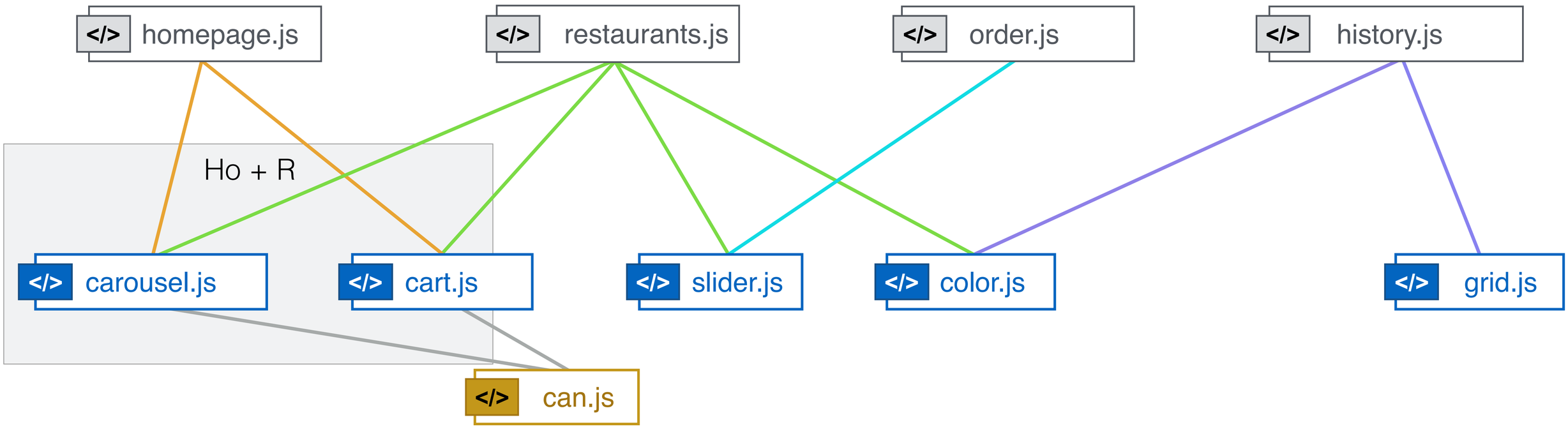


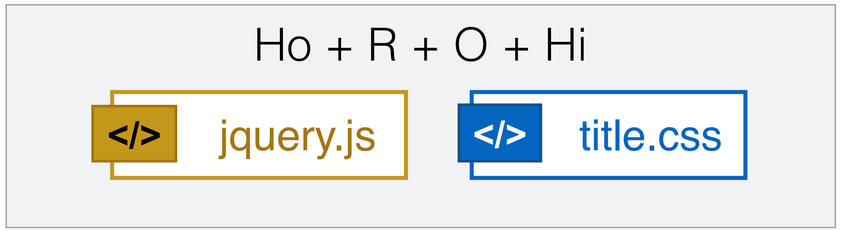
- history.js
  - title.css
  - color.js
  - grid.js

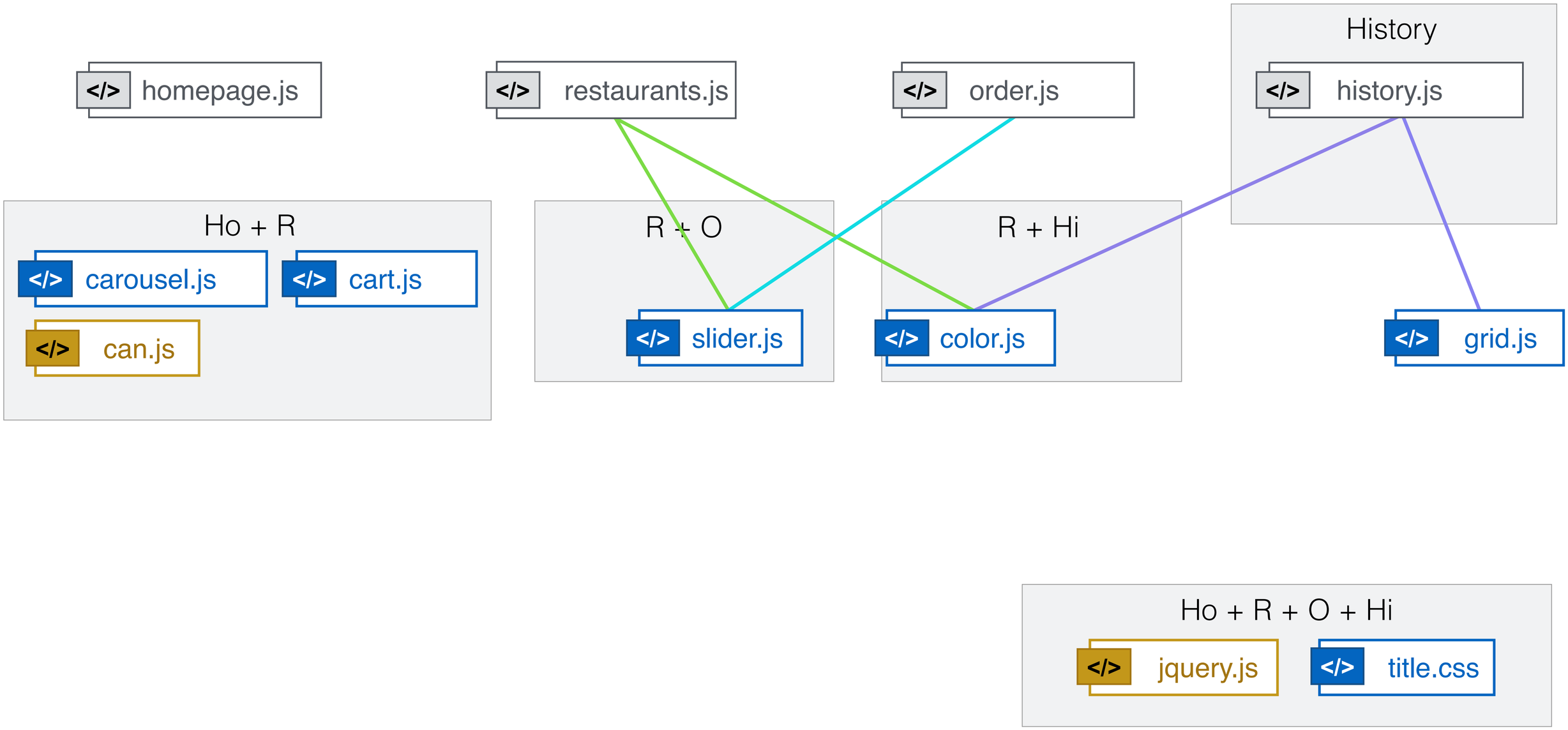
Multi-Page Statically Loaded




homepage.js
restaurants.js
order.js
history.js
carousel.js
cart.js
slider.js
color.js
title.css
grid.js
can.js
jquery.js
Ho + R + O + Hi


homepage.js
restaurants.js
order.js
history.js
Ho + R
carousel.js
cart.js
slider.js
color.js
grid.js
can.js


Ho + R + O + Hi
jquery.js
title.css


homepage.js
restaurants.js
order.js
History
history.js
Ho + R
carousel.js
cart.js
can.js
R + O
slider.js
R + Hi
color.js
grid.js
Ho + R + O + Hi
jquery.js
title.css

steal js

Homepage

</> homepage.js

Restaurants

</> restaurants.js

Order

</> order.js

History

</> history.js

</> grid.js

Ho + R

</> carousel.js

</> cart.js

</> can.js

R + O

</> slider.js

R + Hi

</> color.js

Ho + R + O + Hi

</> jquery.js

</> title.css

steal js

Homepage

homepage.js

Restaurants

restaurants.js

Order

order.js

History

history.js

grid.js

Ho + R

carousel.js

cart.js

can.js

R + O

slider.js

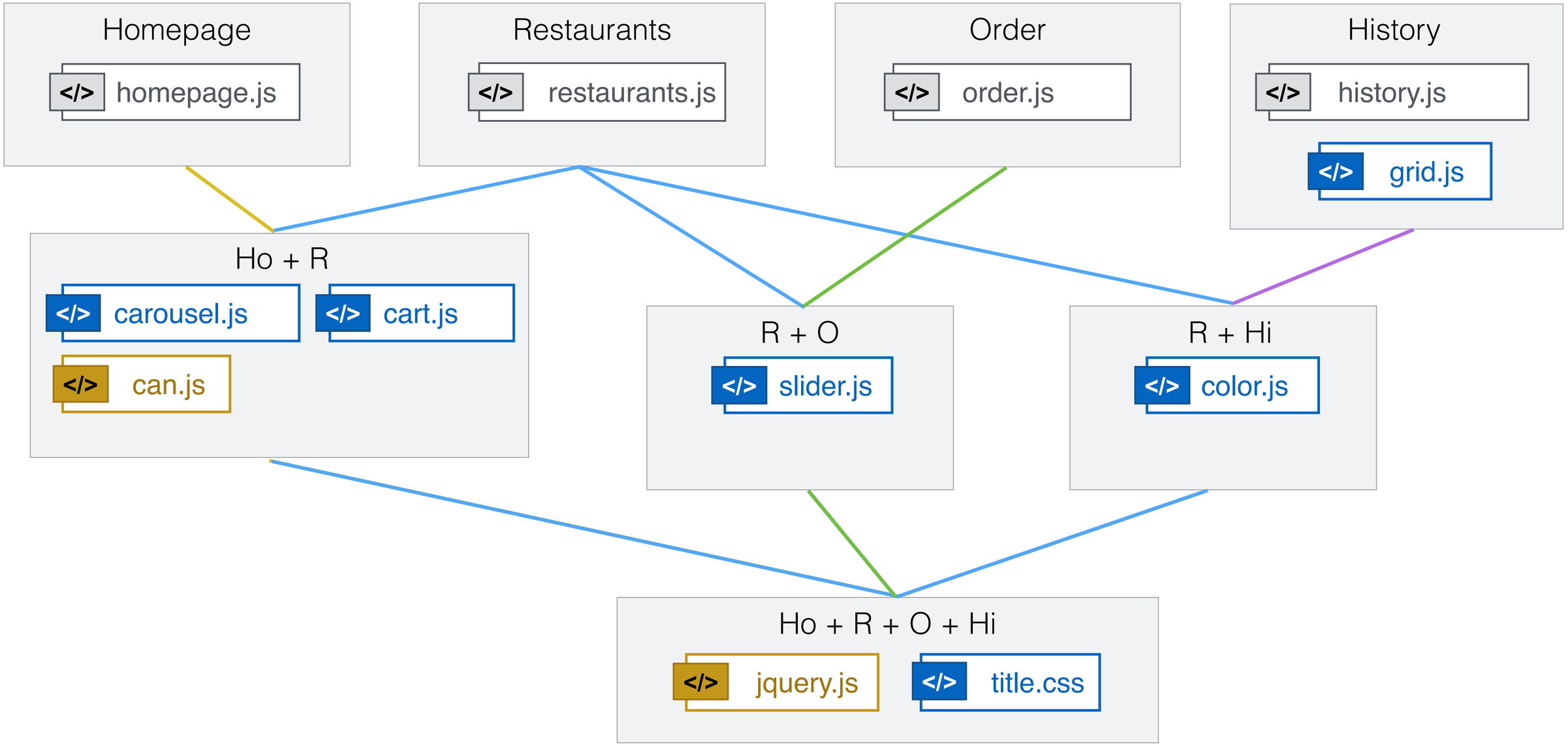
R + Hi

color.js

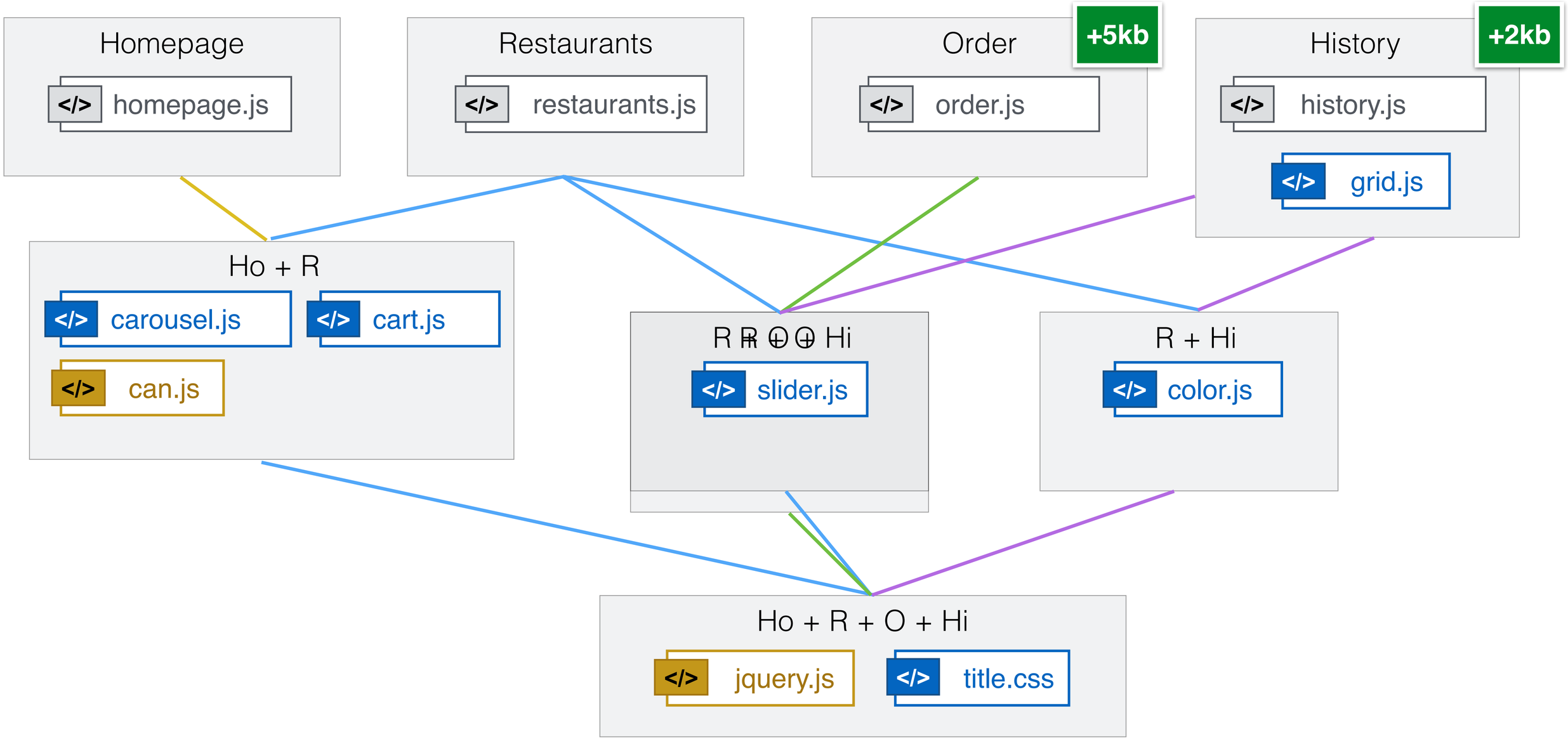
Ho + R + O + Hi

jquery.js

title.css


Homepage
homepage.js
Restaurants
restaurants.js
Order
order.js
History
history.js
grid.js
Ho + R
carousel.js
cart.js
can.js
R + O
slider.js
R + Hi
color.js
Ho + R + O + Hi
jquery.js
title.css

stealjs


Homepage
homepage.js
Restaurants
restaurants.js
Order
+5kb
order.js
History
+2kb
history.js
grid.js
Ho + R
carousel.js
cart.js
can.js
R R O O Hi
slider.js
R + Hi
color.js
Ho + R + O + Hi
jquery.js
title.css

main.stache

```
{{#eq route.page "home"}}
  <can-import from="app/home/home">
    {{#isResolved}}
      <app-home></app-home>
    {{/isResolved}}
  </can-import>
{{/eq}}
{{#eq route.page "restaurants"}}
  {{^slug}}
     <can-import from="app/restaurant/list/list">
      {{#isResolved}}
        <app-restaurant-list></app-restaurant-list>
      {{/isResolved}}
    </can-import>
  {{else}}
```

10. Server Side Rendering

Loading jbmillini@gmail.com…

Demo

Benefits

- SEO
- Perceived Performance

Single Context Virtual DOM

- Everything runs in a single node context.
- Using a very lightweight DOM.
- Everything connected to render lifecycle

main.stache

```
{{#eq route.page "home"}}
  <can-import from="app/home/home">
    {{#isResolved}}
      <app-home></app-home>
    {{/isResolved}}
  </can-import>
{{/eq}}
{{#eq route.page "restaurants"}}
  {{^slug}}
     <can-import from="app/restaurant/list/list">
      {{#isResolved}}
        <app-restaurant-list></app-restaurant-list>
      {{/isResolved}}
    </can-import>
  {{else}}
```

lib/app.js

```
steal.import("app/main.stache!can/view/autorender/system",
             "app/layout.stache!").then(function([ main, layout ]){

  express().use("/", function(req, res){

    var pathname = url.parse(req.url).pathname;
    main.renderNode(pathname).then(function(renderData){

      var html = layout.render({
        content: renderData.html,
        inlineCache: renderData.data
      });
      res.send(html);
  });
});
```

layout/main.stache

```
<html>
  <head>
    <title>Place My Order</title>
  </head>
  <body>
    {{{content}}}
    <script {{inlineCache}}></script>
    {{#if isProd}}
      <script src="/node_modules/steal/steal.production.js"></script>
    {{else}}
      <script src="/node_modules/steal/steal.js"></script>
    {{/if}}
  </body>
</html>
```

place-my-order/restaurant/list.js

```
export const ViewModel = Map.extend({
  states: { get() { return State.findAll({}); } }, …
});

export default Component.extend({
  tag: 'app-restaurant-list',
  viewModel: ViewModel,
  template,
  events: {
      inserted: function(){
            var statesPromise = this.attr("states");
            this.scope.attr("@root").pageData( statesPromise );
      }
  }
});
```

Techniques

- Duplicate Code

  +0 Maintainability +10 Speed

- Headless Browser

  +9 Maintainability +5 Speed

- Single Context Virtual DOM

  +8 Maintainability +8 Speed

Thanks - @bitovi