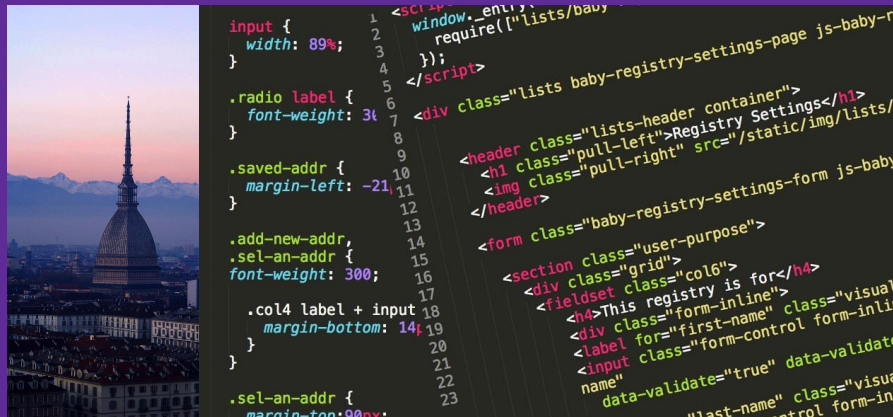


10 March 2016 - London

An introduction to Ember JS for building *ambitious* web applications

by Daniela Remogna

Senior Software Engineer



Web Site: www.fimietta.it



Twitter: [@fimietta](https://twitter.com/fimietta)



Github: [fimietta](https://github.com/fimietta)



Linked in: www.linkedin.com/in/fimietta

Something about me

I'm from from **Turin**, *Italy*

I moved to London one year and a half ago for working as *front end developer* in **Workshare** (an *online document collaboration system similar to Dropbox*)

Currently I'm a *senior software engineer* at **CrowdStrike** where I'm involved in the development of a *real time anti breach, virus and malware platform*.

I'm an *engineer*, I got a master degree in computer science at the *Politecnico of Turin*

In Italy I worked for an automotive company - **Magneti Marelli** as software architect and for an e-learning company - **Docebo** as full stack developer.



Something about me (2)

I am really addicted to front end technologies such as *javascript* and *its frameworks*.

I'm continuously studying patterns for building robust *web applications*.



In my spare time I like cooking italian food, travelling and hiking around the world.

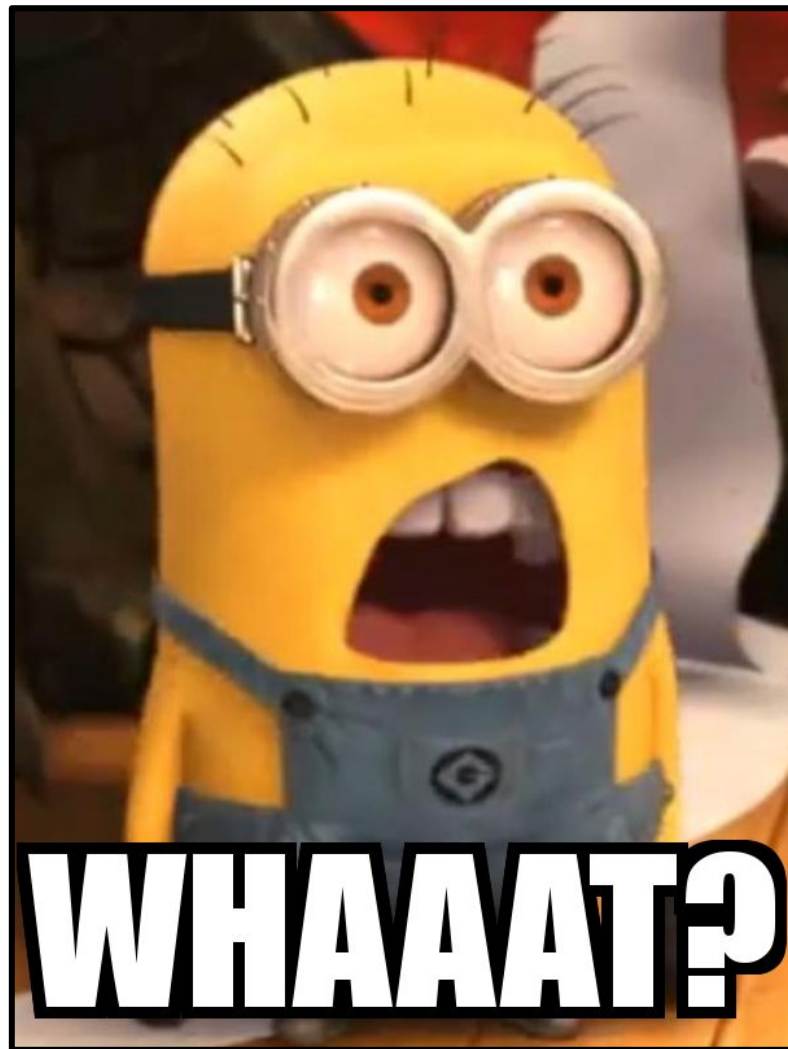


Introduction to Front End Development

When I have to tell which is my job...

I am a Front End Developer.

....this is the people reaction....



I am a Front End Developer - What does it mean?

Everyday we are using *web applications* via our browser like *amazon*, *dropbox*, *facebook*, *barclays* for internet banking, *spare room*, *gum tree* and so on.

“A front end developer builds the **user interface** of an application. “

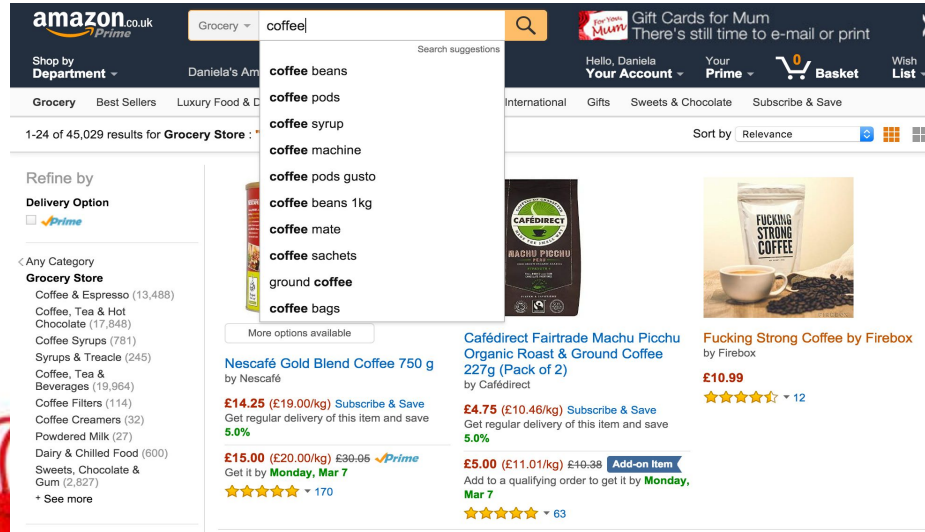
On a web site like amazon, the front end developer is responsible for building components like:

SEARCH FORM - WRITE A REVIEW - VIEW THE RELATED PRODUCTS

Each component of the user interface usually has an **html/css** part, then a **javascript part** for handling the interaction with the user or for retrieving the data from the server.

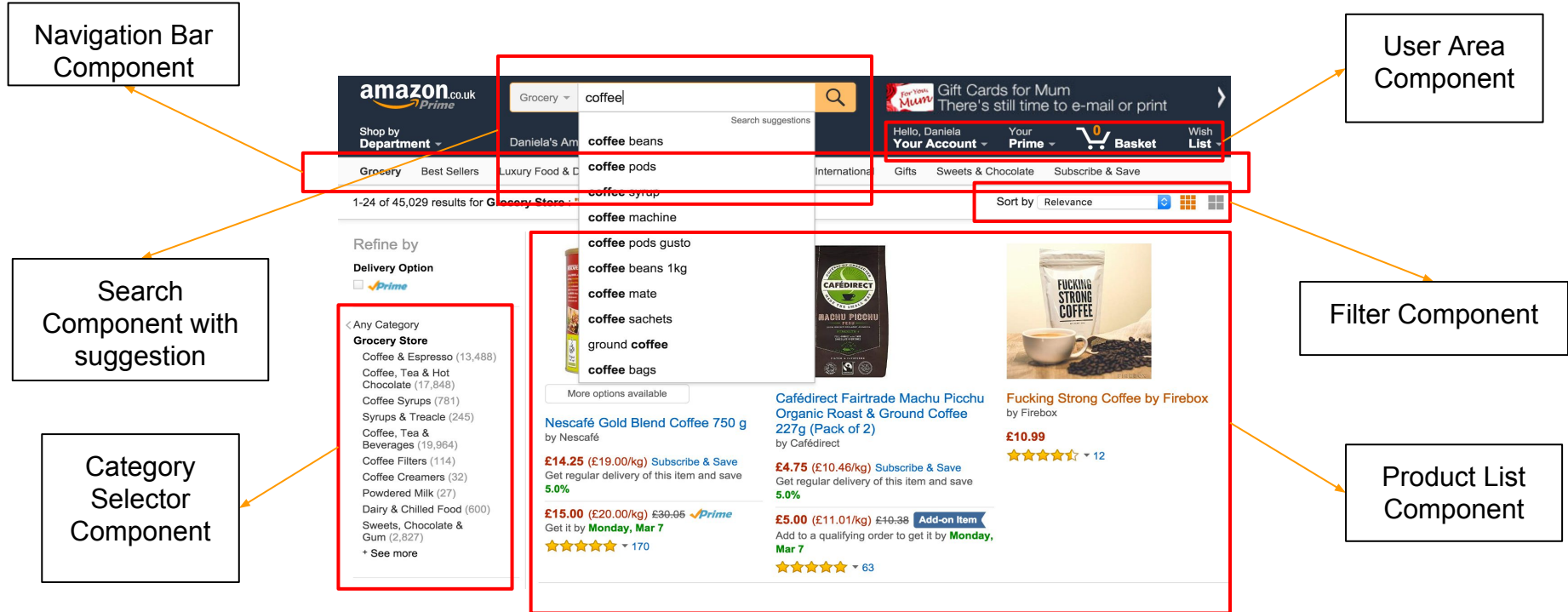
How a “normal” user sees the amazon web site

A place where people can spend a lot of money for buying new stuff :-)

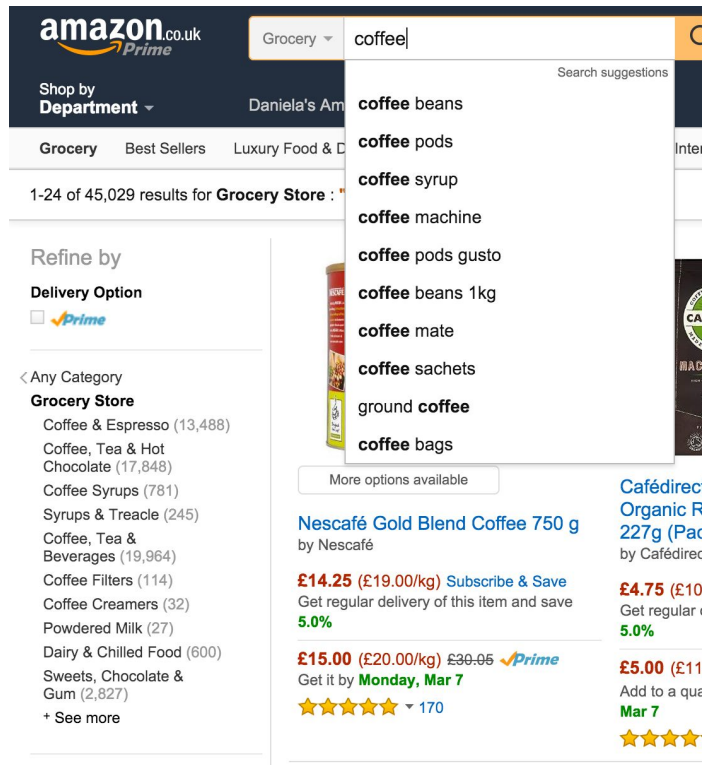


How a front-end developer (sees) the amazon site

A front end developer sees the amazon site as a collection of web components :)



There is also a back end developer



A web site without data is NOTHING!!

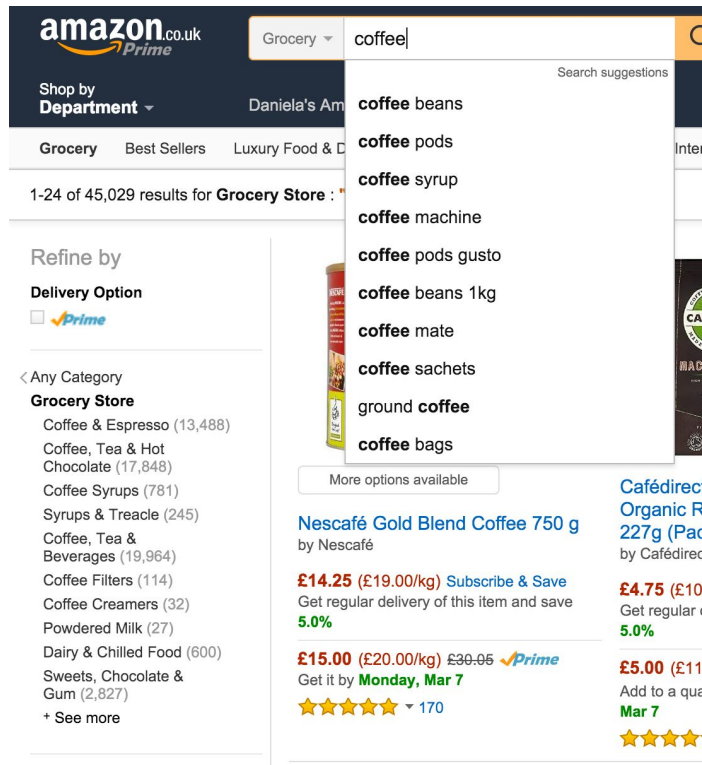
All the data you see on a web site are stored persinsently somewhere (hopefully on a database).

The front end developer is responsible for asking data to the back end.

The back-end developer GIVES the answers with data!

Usually he/she works with the database.

There is also a back end developer



Let's focus on the search component

The **search component** asks to the server the *products list* containing the word **coffee** by invoking a proper URL

(like `www.mydomain.com/search?name=coffee`)

The **back end developer** wrote a piece of code that reads the name from the url and then search the products matching the word coffee in the database and return the list to the client (search component).

The component wait for the response, then when it's ready, it updates the search suggestions with the results.

Where can I learn how to code?

You don't know, but all the knowledge it's in your pockets thanks to internet.

Online:

- CodeSchool <https://www.codeschool.com/>
- Code Academy <https://www.codecademy.com/>
- Khan Academy <https://www.khanacademy.org/computing/computer-programming/html-css-js>
- Code Avengers <https://www.codeavengers.com/>

Offline:

- Go to **Codebar** - <https://codebar.io/> - <http://tutorials.codebar.io/>
- Attend to Meetup about Javascript, Web Development, Ember
- Attend to event organized by **Skills Matter** <https://skillsmatter.com/>
- Founders & Coders - <http://www.foundersandcoders.com/>



Meetup in London

- <http://www.meetup.com/London-Emberjs-User-Group/>
- <http://www.meetup.com/AngularLabs/>
- <http://www.meetup.com/js-monthly-london/>
- <http://www.meetup.com/Front-Engineers-London/>
- <http://www.meetup.com/Ladies-of-Code-UK/>
- <http://www.meetup.com/London-JavaScript-Community/>
- <http://www.meetup.com/London-React-User-Group/>
- <http://www.meetup.com/Women-Who-Code-London/>
- <http://www.meetup.com/Geek-Girls-Carrots-UK>



Ember

Introduction to the concept of Framework

A framework:

- makes life easier for a developer.
- help to structure your code in order to address a particular problem
- allow developers to focus on business logic of the application

The most famous Javascript framework are:



What is Ember JS

- A Javascript framework for creating **ambitious** web applications
- Based on a model-view-controller (MVC)
- Designed for Single Page Application
- Provides standard application architecture
- Follow Convention Over Configuration Pattern
- Use Handlebars as Template Engine with TWO-WAY data binding



Who is using Ember?

YAHOO!

GROUPON

NETFLIX

KICKSTARTER

 EatWith

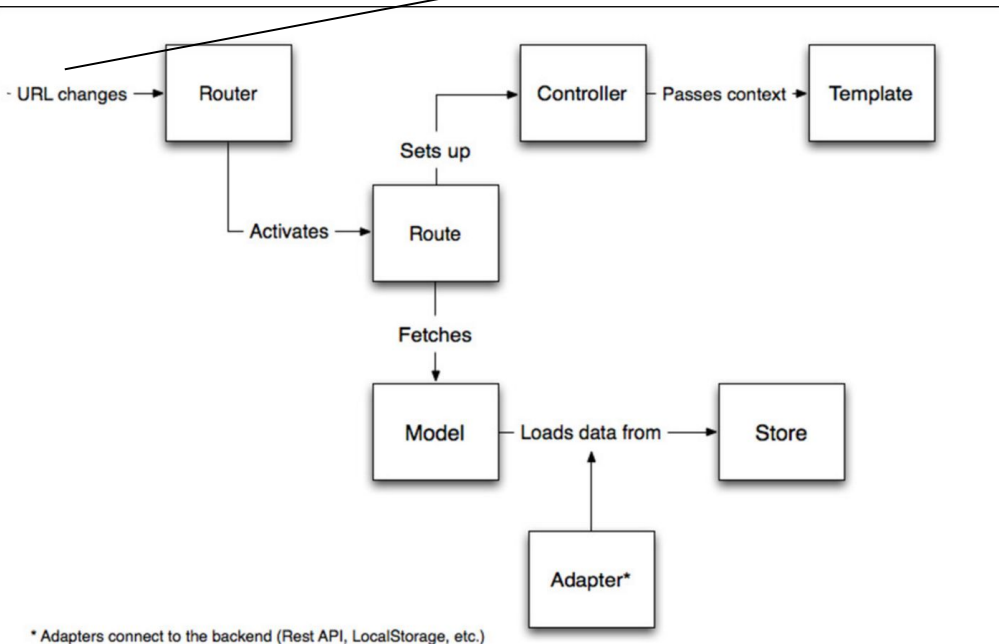
LinkedIn

runtastic
makes sports funtastic

code school

<http://emberjs.com/ember-users/>

Ember Workflow



* Adapters connect to the backend (Rest API, LocalStorage, etc.)

www.shop.com/products/book

www.shop.com/products/book/crime-fiction

www.shop.com/offers

From an **URL**, the router translates an URL to a specific **route**. So after parsing the URL the router activates a **route**.

A **Route** is responsible for:

- *fetching* a model from a server
- *providing the model* to the controller

The route *sets up a controller* that decorates the model by adding new properties. In the controller you can also specify the **actions** for handling the **user interactions**.

And finally the **template is loaded** and shown to the user.

Ember 5 Key Concepts

1

Router

2

Route

3

Model

4

Template

5

Controller

Everything starts from an **URL** - in Ember **URLs** on a web application are really important. They are like a “map” that drive your application to the specific page via a *route*!

Ember **Router** maps the current URL to one or more **route** handlers.

The **route** is responsible for giving the **model (data)** to the *template*.

The **template** defines the *layout* and the *user interface of a page/component*. Template in Ember support *two-way data bindings*, so it will auto update when the related model changes!

The **controller** in Ember decorates the model object. Properties specified in the controller can be accessed by a template as if they were declared in the model.

Question??

Given an URL like:



How Ember knows from an **url** which is the related **route**, **controller** and **template**? How these blocks are connected each other?

How does Ember know that??



Convention Over Configuration Pattern in Ember

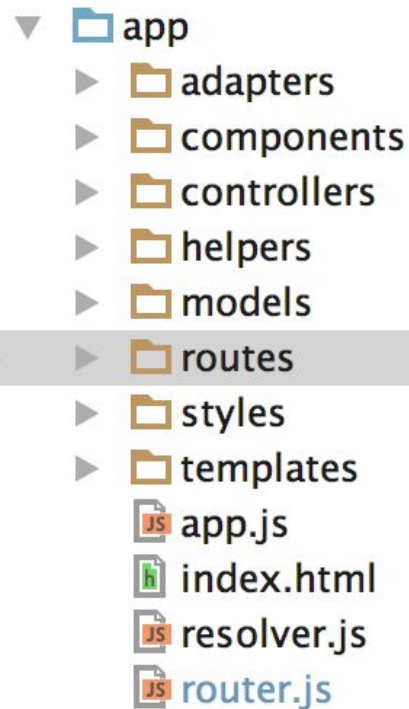
Ember knows the proper route, controller and template thanks to

“a **naming convention** and a **standard application folder structure**”

`http://www.shop.com/products`

- `app/controllers/products.js`
- `app/routes/products.js`
- `app/template/products.hbs`

Folder structure of
an ember
application!



Demo Time!!



Catalog your music collection with Ember.js

Add

Pearl Jam

Led Zeppelin>

Kaya Projecttt

Foo Fighters

Radiohead

Red Hot Chili Peppers

DetailsSongs

Rating↓Rating↑Title↓Title↑

Q

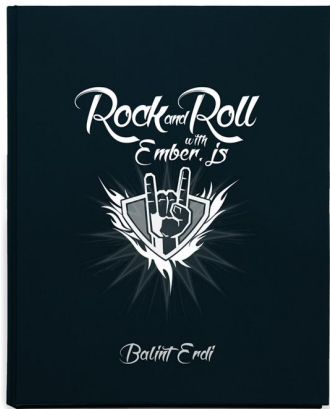
Add

Achilles Last Stand

Black Dog

Immigrant Song

Resources about Ember



Reading this fabulous book "***Rock and Roll with Ember.js***" written by Balint Erdi.

You will learn step by step how to develop the application shown on the demo.

<http://balinterdi.com/rock-and-roll-with-emberjs/>

Other Links:

- <http://emberweekly.com/>
- <http://emberjs.com>
- <https://www.codeschool.com/courses/try-ember>



Thank YOU!

Q & A