

Web application design patterns

Amy Ruffle

CodeHub Javascript101, April 2016

What is a design pattern?

“a reusable solution that can be applied to commonly occurring problems in software design”

Addy Osmani

[Learning Javascript Design Patterns](#) by Addy Osmani licenced under [Creative Commons Attribution-Noncommercial-No derivative works 3.0](#)

Design pattern one: the MV* family

- **MVC** = Model-View-Controller
- **MVP** = Model-View-Presenter
- **MVVM** = Model-View-ViewModel

Learning Javascript Design Patterns by
Addy Osmani licenced under
Creative Commons Attribution-Noncom
mercial-No derivative works 3.0

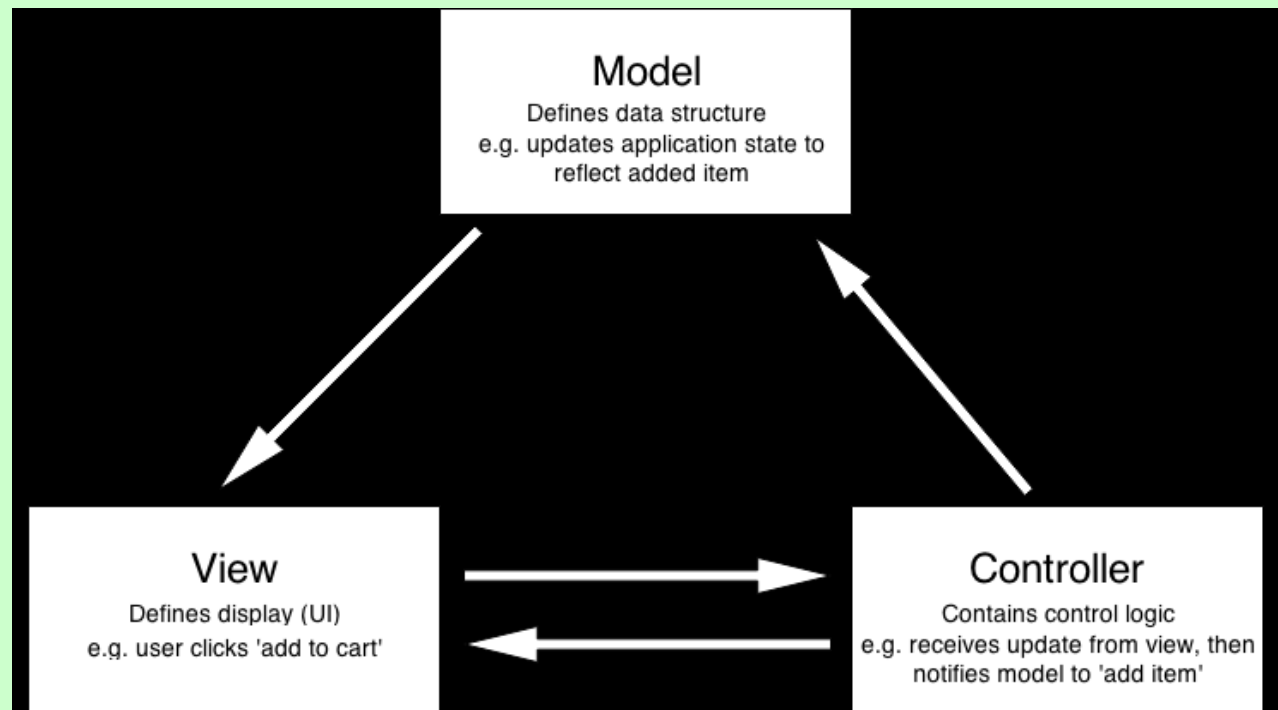


Image from MVC Architecture by Mozilla Contributors licenced under CC-BY-SA 2.5

Implementations of MV*

e.g. Javascript MVC frameworks:

- **AngularJS** - <https://angularjs.org/>
- **Ember.js** - <http://emberjs.com/>
- **Backbone.js** - <http://backbonejs.org/>
- **Sencha** - <https://www.sencha.com/>
- **Kendo UI** - <http://www.telerik.com/kendo-ui>

Comparing two implementations

A hypothetical **Django** approach

Data store: PostGres

App: largely in Python

Model

- models.py

View

- views.py
- template (HTML, CSS, vanilla Javascript, and Django template language)

Controller

- urls.py
- vanilla Javascript

A hypothetical **Ember.js** approach

Data store: IndexedDB

App: rich client-side app, largely in Javascript

Model

- Adapter (converts to JSON) and serializer (converts to ember.js data record)
- Data store (representation of the data structure)

View

- a 'route'
- a template (HTML, CSS with Handlebars template language)

Controller

- a 'controller'

Design pattern two: Flux

- Components (Dispatcher, Store, View) are not that different to MVC
- But key difference is that **data only flows in one direction**, making it easier to manage an application as it gets bigger and more complicated.
- Recommend watching talk at <https://facebook.github.io/flux/docs/overview.html>

Implementations of Flux

e.g.

- **Fluxxor** - <http://fluxxor.com/what-is-flux.html>
- **Alt** - <http://alt.js.org/>

Also, **Redux** is “a predictable state container for JavaScript apps” which uses some of the ideas of Flux - <https://github.com/reactjs/redux>

Helpful resources

On design patterns generally

[Learning Javascript Design Patterns](#) by Addy Osmani licenced under Creative Commons Attribution-Noncommercial-No derivative works 3.0

On MVC

[Modern web app architecture](#) by Mozilla Contributors licenced under CC-BY-SA 2.5

[MVC Architecture](#) by developer.chrome.com licenced under CC-By 3.0 license

Chapter 'Javascript MV* Patterns' in [Learning Javascript Design Patterns](#) (as above)

On Ember

[Modern web app architecture](#) by Mozilla Contributors licenced under CC-BY-SA 2.5

On Django

[Writing your first Django app tutorial](#)

[Django FAQ on Django as an MVC framework](#)

On LokiJS

[LokiJS site](#) including presentation and slides from CorkDev