Architecture and Main Characteristics

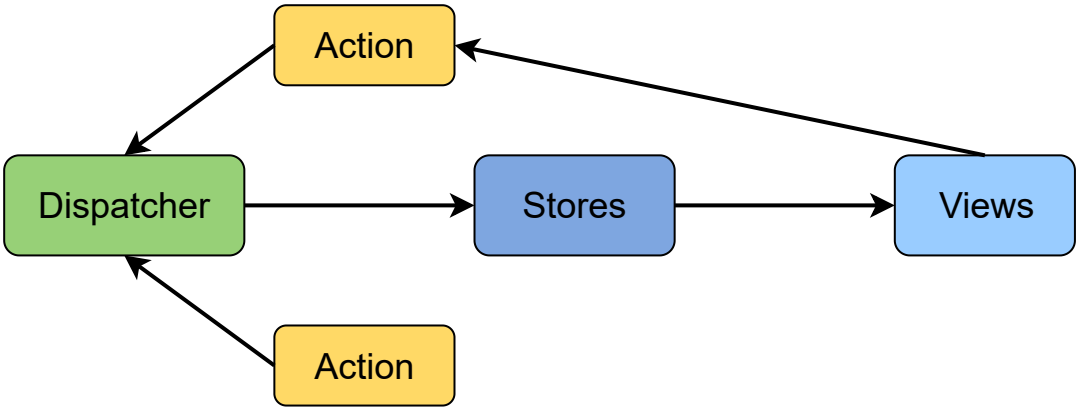
Introducing Flux, its architectural design pattern for building user interfaces, commonly used with React to make code simpler and its main characteristics.

WE'LL COVER THE FOLLOWING

Flux: What is it?

Flux is an architectural design pattern for building user interfaces. It was introduced by Facebook at their F8 conference. Since then, lots of companies adopted the idea and it seems like a good pattern for building front-end apps. Flux is very often used with React and is simple and really flexible. The pattern helps to create apps faster and at the same time keeps the code well organized.

Flux: Architecture and Main Characteristics



The main actor in this pattern is the *dispatcher*. It acts as a hub for all the events in the system. Its job is to receive notifications that we call *actions* and pass them to all the *stores*. The store decides if it is interesting or not and reacts by changing its internal state/data. That change is triggering re-rendering of the *views* which are (in our case) React components. If we have

rendering of the *views* which are (in our case) React components. If we have

to compare Flux to the well known MVC we may say that the store is similar to the model. It keeps the data and its mutations.

The actions are coming to the dispatcher either from the views or from other parts of the system, like services. For example, a module that performs a HTTP request. When it receives the result it may fire an action saying that the request was successful.

Now that we know what Flux is, we can move on to see how a Flux architecture can be implemented. As every other popular concept Flux also has some variations. Very often to understand something we have to implement it. In the next few lessons, we will create a library that provides helpers for building the Flux pattern.