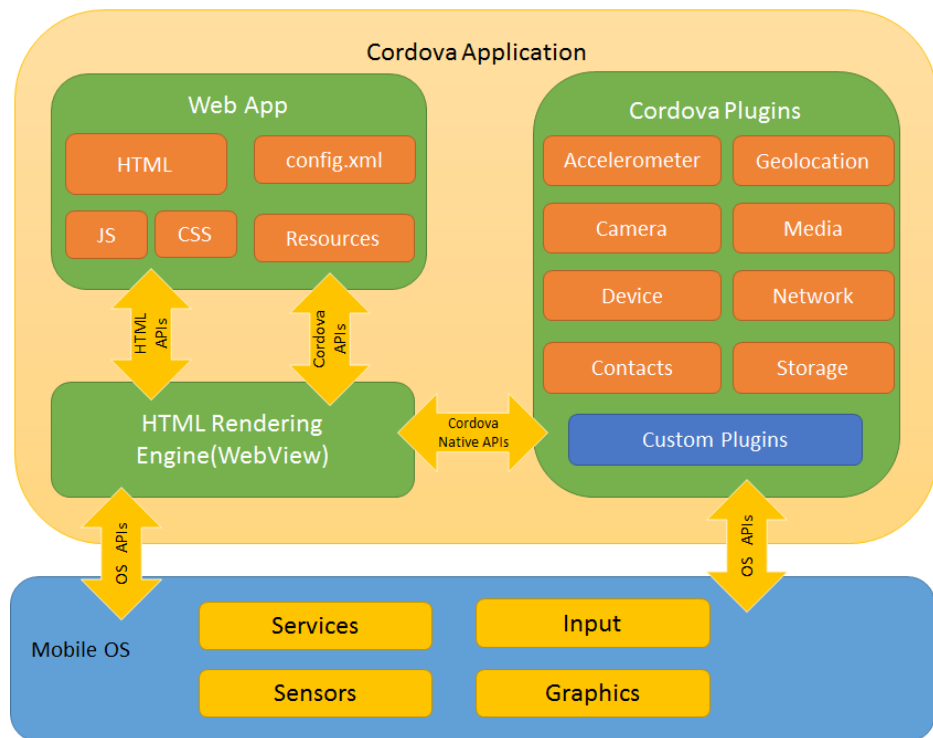




cordova - react - redux

the mobile application

Cordova



Web View

Motore di rendering HTML.

Web App

L'applicazione viene implementata utilizzando le tecnologie web come HTML, CSS e JavaScript e viene eseguita in una WebView "wrappata" da un'applicazione nativa.

Plugins

Forniscono un'interfaccia per la comunicazione tra Cordova e i componenti nativi, permettendoti di invocare codice nativo da JavaScript.



React

React è una libreria JS per creare interfacce utente, creata ed utilizzata da Facebook ed Instagram.

Solitamente si pensa a React come la V in MVC.

React è stato creato con lo scopo di risolvere un problema: sviluppare applicazioni di grandi dimensioni il cui stato cambia nel tempo.



React

Gestisce l'aggiornamento dell' UI appena lo stato dell'app cambia (Virtual DOM).

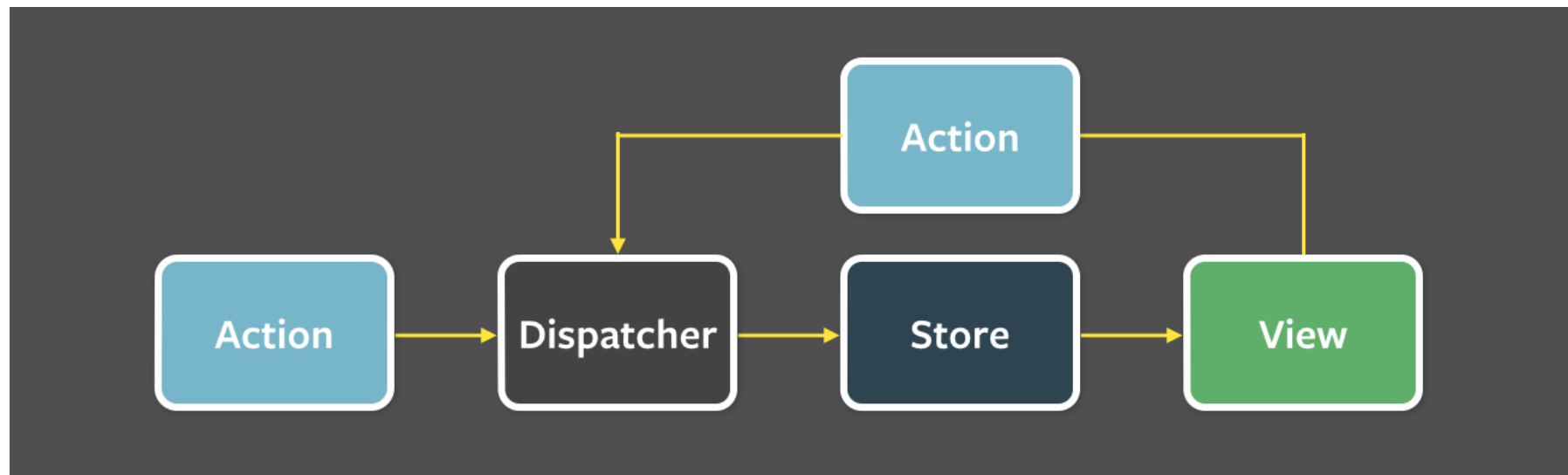
Sviluppo di componenti riutilizzabili, favorendo il riutilizzo del codice, il testing ed il debugging.



Flux

Flux è un pattern architetturale utilizzato da Facebook nello sviluppo client-side delle web-app.

Utilizzato in complemento a React, fornisce un data-flow unidirezionale.



Redux

Redux è una delle implementazioni del pattern Flux.

Cerca di rendere i cambiamenti dello stato prevedibili imponendo delle restrizioni su come essi possono verificarsi, seguendo tre principi fondamentali:



Single source of thruth

Lo stato dell'intera applicazione è conservato in un singolo oggetto "store".

```

next state ▾ Object
  ▾ drawer: Object
    isOpen: false
    ▶ __proto__: Object
  ▾ eventFilters: Object
    city: -1
    selected: "FILTER_0"
    ▶ __proto__: Object
  ▾ events: Object
    ▶ data: Array[7]
    ▶ favourites: Array[1]
    favouritesFilterActive: false
    layout: "GRID_VIEW"
    ▶ __proto__: Object
  ▾ network: Object
    online: false
    ▶ __proto__: Object
  ▾ news: Object
    ▶ data: Array[3]
    ▶ __proto__: Object
  ▾ routing: Object
    ▶ locationBeforeTransitions: Object
    ▶ __proto__: Object
  ▾ screen: Object
    height: 640
    width: 384
    ▶ __proto__: Object
  ▾ tabs: Object
    ▶ eventDetail: Object
    ▶ home: Object
    ▶ __proto__: Object
    ▶ __proto__: Object
  
```



State is read-only

L'unico modo per modificare lo stato è quello di emettere una “action”, un oggetto che descrive l'azione da eseguire.

```
1 // Action constants
2 export const OPEN_DRAWER = "OPEN_DRAWER";
3 export const CLOSE_DRAWER = "CLOSE_DRAWER";
4 export const TOGGLE_DRAWER = "TOGGLE_DRAWER";
5
6 // Action creators
7 export const openDrawer = () => {
8   return {
9     type: OPEN_DRAWER
10  }
11 }
12
13 export const closeDrawer = () => {
14   return {
15     type: CLOSE_DRAWER
16   }
17 }
18
19 export const toggleDrawer = () => {
20   return {
21     type: TOGGLE_DRAWER
22   }
23 }
```



Pure Functions

Per specificare il modo in cui lo stato viene modificato in risposta ad una azione, devi scrivere funzioni pure.

```
12 const drawer = (state = defaultState, action) => {
13   switch (action.type) {
14     case OPEN_DRAWER:
15       return {
16         ...state,
17         isOpen: true
18       }
19     case CLOSE_DRAWER:
20       return {
21         ...state,
22         isOpen: false
23       }
24     case TOGGLE_DRAWER:
25       return {
26         ...state,
27         isOpen: !state.isOpen
28       }
29     default:
30       return {
31         ...state
32       }
33   }
34 }
35
36 export default drawer
```



Sviluppo

Partiamo da uno scheletro di applicazione che abbiamo sviluppato e rilasciato su GitLab.

```
#> npm install -g cordova
#> git clone https://gitlab.com/wish-op/skeleton_react_mobile_application.git
#> cd skeleton_react_mobile_application
#> npm install
#> cordova platform add android
#> npm run prepare
#> npm start
#> npm run android
```



La struttura

```

francesco@wish-op: ~/test/skeleton_react_mobile_application
francesco@wish-op:~/test/skeleton_react_mobile_application$ tree -d -L 3 -I node_modules
.
├── app
│   ├── actions
│   ├── components
│   ├── containers
│   ├── reducers
│   └── store
├── hooks
├── platforms
│   └── android
│       ├── assets
│       ├── cordova
│       ├── CordovaLib
│       ├── libs
│       ├── platform_www
│       ├── res
│       └── src
├── plugins
│   └── cordova-plugin-whitelist
│       ├── doc
│       └── src
└── www
    ├── css
    │   └── bootstrap
    └── libs
    
```

24 directories
francesco@wish-op:~/test/skeleton_react_mobile_application\$



Live coding...



Link Utili

Cordova: <https://cordova.apache.org/docs/en/latest/>

Node/Npm: <https://docs.npmjs.com/>

React: <https://facebook.github.io/react/docs/getting-started.html>

Flux: <https://facebook.github.io/flux/docs/overview.html#content>

Redux: <http://redux.js.org/>

WebPack: <https://webpack.github.io/docs/>

GitLab Repo: https://gitlab.com/wish-op/skeleton_react_mobile_application.git



WISH-OP

“Go and spread the free software.

HAPPY HACKING!”

RMS

info@wish-op.com

347 76 99 240 - 0824 33 02 04

