
Lab 2 - The Cordova CLI & MFP v8.0 Cordova Plugin

This lab will use the Cordova CLI to add the MobileFirst Cordova plugin. These are the standard steps required to add plugins to an application. Adding the MobileFirst plugin allows the application to use the MobileFirst platform features.

General Steps:

1. Within the IBM Employee application directory, run Cordova CLI command to add a plugin.
2. Add one or more device platforms.
3. Use the MobileFirst CLI to preview the application to ensure the plugin was successfully added.

Steps

1. Change context into the MobileFirst project.

```
cd IBMEmployeeApp
```

2. **Add** the iOS platform, run the following command cordova platform add ios

```
cordova platform add ios
```

```
Elirans-MacBook-Pro:MadridMFPLab eliran_pro$ cordova platform add ios
Adding ios project...
iOS project created with cordova-ios@4.1.1
```

3. **Run** the cordova plugin add cordova-plugin-mfp

```
cordova plugin add cordova-plugin-mfp
```

mongod	node	bash	node	bash	bash
<pre> Elirans-MacBook-Pro:IBMEmployeeApp eliran_pro\$ cordova plugin add cordova-plugin-mfp Fetching plugin "cordova-plugin-mfp" via npm Installing "cordova-plugin-mfp" for ios Dependent plugin "cordova-plugin-device" already installed on ios. Fetching plugin "cordova-plugin-dialogs" via npm Installing "cordova-plugin-dialogs" for ios Fetching plugin "cordova-plugin-globalization" via npm Installing "cordova-plugin-globalization" for ios cp: no such file or directory: /Users/eliran_pro/Documents/projects/Madrid2016/IBMEmployeeApp/platforms/ios/Employee/main.m If you made changes to your main.m file, manually merge main.m.bak with the main.m file that is provided with IBM MobileFirst Platform Foundation. Elirans-MacBook-Pro:IBMEmployeeApp eliran_pro\$ </pre>					

Note: Running the add plugin command above will add all the required MFP plugin files from npm. This requires a network connection. For more information on the MFP Cordova Plugin, visit: <https://www.npmjs.com/package/cordova-plugin-mfp>

4. **Start** the mobile first server, navigate to the mobile first server installation folder and run the following commands:

For Mac

- Open a new terminal session

```
cd ~/MobileFirst-8.0.0.0
```

```
./run.sh
```

```

Elirans-MacBook-Pro:mfp-server-all-in-one eliran_pro$ ./run.sh
Picked up JAVA_TOOL_OPTIONS: -DwlDevEnv=true
objc[25473]: Class JavaLaunchHelper is implemented in both /Library/Java/JavaVirtualMachines/jdk1.7.0_80.jdk/Contents/Home/jre/bin/java and /Library/Java/JavaVirtualMachines/jdk1.7.0_80.jdk/Contents/Home/jre/lib/libinstrument.dylib. One of the two will be used. Which one is undefined.
Listening for transport dt_socket at address: 10777
Launching mfp (WebSphere Application Server 8.5.5.8/wlp-1.0.11.cl50820151201-1942) on Java HotSpot(TM) 64-Bit Server VM, version 1.7.0_80-b15 (en_US)
[AUDIT   ] CWWKE0001I: The server mfp has been launched.

```

For Windows ???

5. **Run** the mfpdev app preview

```
mfpdev app preview
```

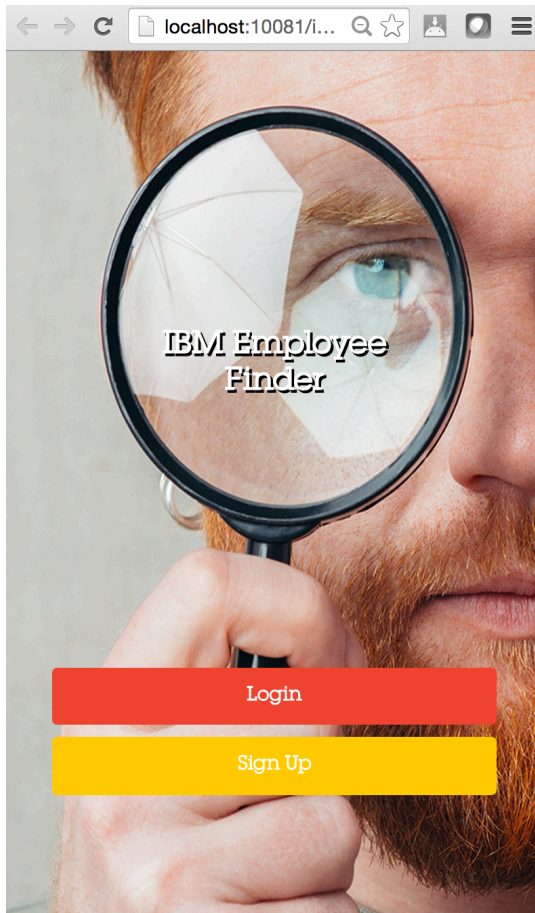
```

Elirans-MacBook-Pro:IBMEmployeeApp eliran_pro$ mfpdev app preview
Verifying server configuration...
? Select how to preview your app: browser: Simple browser rendering
The CLI preview will remove the Content-Security-Policy meta tag during preview to allow the live reload to function in a browser. Please ignore the 'No Content-Security-Policy' error messages from whitelist.js in a browser's debugger console.
[BS] Access URLs:
-----
Local: http://localhost:10081
External: http://10.0.1.5:10081
-----
UI: http://localhost:3001
UI External: http://10.0.1.5:3001
-----
[BS] Serving files from: ./platforms
Launching browser.

```

6. The preview will prompt for using either the simple browser or the mobile browser simulator. Select the simple browser. This will launch the application using the default browser.

```
Verifying server configuration...  
? Select how to preview your app: (Use arrow keys)  
› browser: Simple browser rendering  
mbs: Mobile Browser Simulator
```



Tip: To change the default browser use the `mfpdev config` command.

Summary

This lab enabled the MobileFirst Platform functionality for an existing Ionic/Cordova application by simply issuing the Cordova plugin add command.

If you were unable to complete this lab, you can catch up by running this command: **** You will still need to add the iOS platform and the mfp cordova plugin manually since the plugin folder is not committed to the git repo**

```
git checkout -f step-2
```