



**Mobile User Experience Monitoring**  
**Hybrid Apps – Cordova**

# Contents

Environment .....	3
Bundling eG Mobile Agent with Application.....	3
Installing eG Cordova Agent for Android Applications .....	3
Build and run the android application .....	4
Installing eG Cordova Agent for iOS Applications.....	5
Build and run the iOS application .....	6
Troubleshooting for JS Error in ionic angular applications .....	6

# Environment

Cordova CLI	8+
Cordova Platforms	android 9+
NodeJS	v10+
NPM	6+
OS	Windows 10/ IOS

## Bundling eG Mobile Agent with Application

The following steps are required to bundle the 'eG Mobile Agent' into Hybrid apps (Cordova) to activate user experience monitoring. Cordova apps monitored using following components.

1. eGCordova Agent.
2. eGJS Agent.

## Installing eG Cordova Agent for Android Applications

- 1) Add 'eG-Hybrid' Apps Component into eG Manager Console and copy the token tied with this component.
  - a) Run the below command from command prompt from application's root folder by replacing the values for ANDROID\_APP\_TOKEN and MANAGER\_URL parameters. Here <PATH> refers full path of the extracted plugin zip file.

```
cordova plugin add <PATH>@eginnovations/eginnovations-cordova-plugin --variable  
ANDROID_APP_TOKEN="xxxx-xxxxxx-xxxxxx" --variable MANAGER_URL="http://<host-ip>:7077" --variable  
LOG_LEVEL="6"
```

- b) Copy and paste the highlighted content as shown in the below screen shot (Fig.2) into header section of mobile application's index.html page. Sample content as below

Fig.1

Add Component ⓘ

Back

Component Information

Category

All

Component type

Mobile RUM

Nick name

CordovaAndroid

Mobile RUM collector

Default Collector

Remote agent

172.16.14.188

Application type

☐ Native ☒ Hybrid

Add

Fig.2

Cordova

Capacitor

React

Manual

NPM

01

Download the PDF and follow the steps to inject eG Hybrid Cordova Agent into your mobile apps

Download

02

Download eG Agent for Cordova

Download

03

Include this line into applications index.html to inject eG JS Agent into your mobile apps

Copy to Clipboard

```
<!-- RUM Header -->
<script charset='UTF-8' type='text/javascript'>
window[egrum-start_time] = new Date().getTime();
window[Site_Name] = '55026b01-69ec-46b9-8d5d-f38f42bfca90-1644506355557';
window[beacon-url] = 'http://172.16.14.188:7077';
```

## Build and run the android application

Clean the project and build the application. After successful build, run the application in an emulator or mobile device and login to eG manager application to start seeing data.

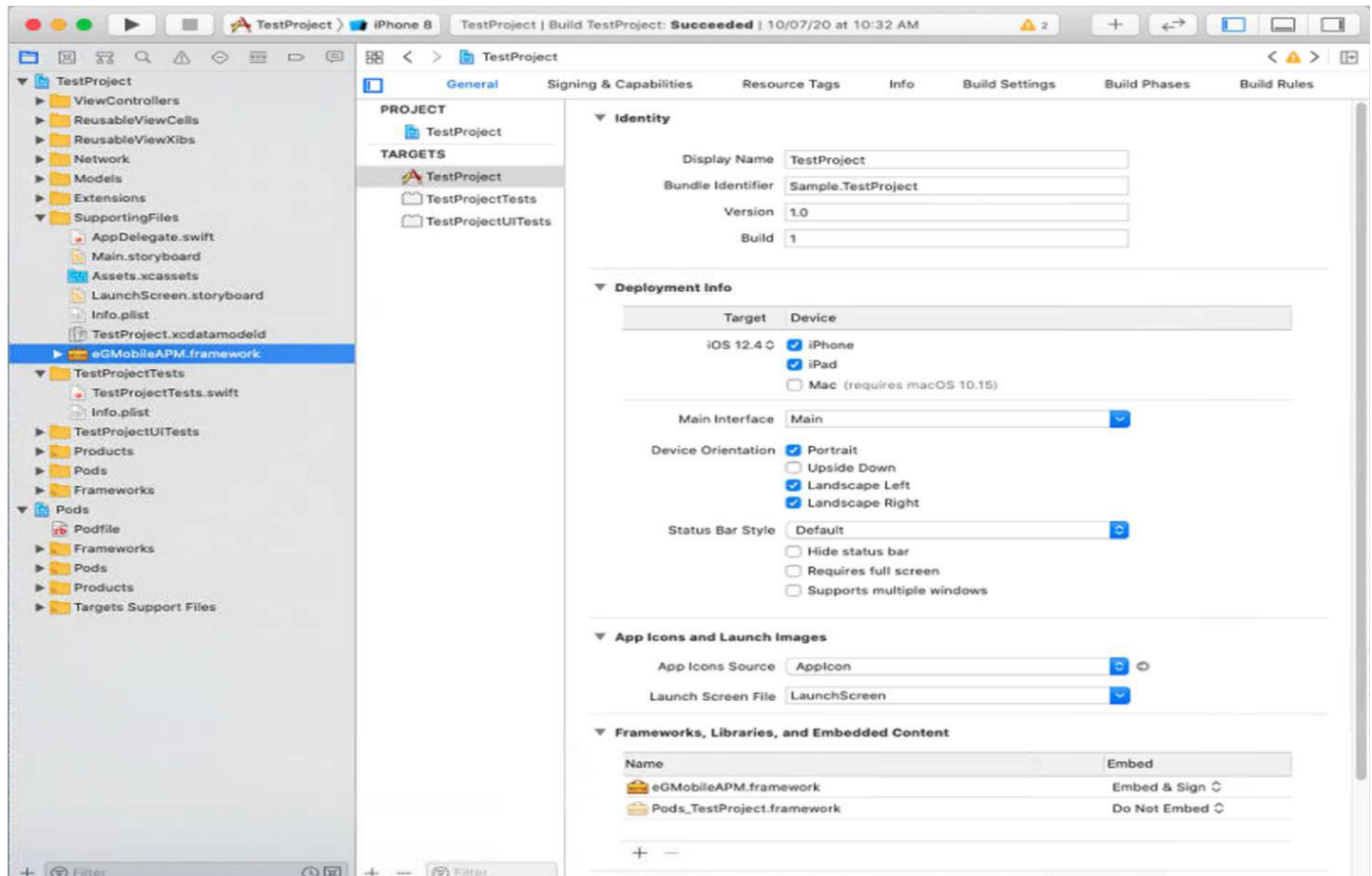
# Installing eG Cordova Agent for iOS Applications

- Add 'eG-Hybrid' Apps Component into eG Manager Console and copy the token tied with this component.
- Run the below command from command prompt from application's root folder by replacing the values for IOS\_APP\_TOKEN and MANAGER\_URL parameters.

```
cordova plugin add <PATH>@eginnovations/eginnovations-cordova-plugin --variable  
ANDROID_APP_TOKEN="xxxx-xxxxxx-xxxxxx" --variable MANAGER_URL="http://<host-ip>:7077" --variable  
LOG_LEVEL="6"
```

- Copy and paste the highlighted content as shown in the (Fig.2) into header section of mobile application's index.html page.
- In the **General** settings of your project '**Embed & Sign**' the eGMobileAPM Framework in Framework, Libraries, and Embedded Content.

Fig.3



# Build and run the iOS application

Clean the project and build the application. After successful build, run the application in an simulator or mobile device and login to eG manager application to start seeing data.

## Troubleshooting for JS Error in ionic angular applications

Add the following code in app.module.ts file in the application.

Ref: <https://angular.io/api/core/ErrorHandler>

```
import { ErrorHandler } from '@angular/core';
declare var EGRUM: any;
@Injectable()
export class EGRUMErrorHandler implements ErrorHandler {
  handleError(err: any): void {
    EGRUM.onerror({
      message: err.message,
      filename: err.filename,
      lineNumber: err.lineNumber,
      columnNumber: err.columnNumber,
      stack: err.stack,
      meta: { type: "Internal Error." }
    });
  }
}

@NgModule({
  declarations: [AppComponent],
  entryComponents: [],
  imports: [BrowserModule, IonicModule.forRoot(), AppRoutingModule],
  providers: [
    {provide: ErrorHandler, useClass: EGRUMErrorHandler}
  ],
  bootstrap: [AppComponent]
})
```