**\*How to use AppCAMOFramework**

Please refer to the changed AppCAMO-ios application method below.

**Step1. Check the Bitcode setting value.**

 1. Run Xcode.

 2. Search for "Bitcode" in the Build Settings item in

 your project and targets.

 Check the Bitcode setting you are using in Xcode.

**Step2. Unzip the AppCAMOFramework.**

 1. unzip ios\_framework\_20230118.zip

 When you unzip, "framework\_bitcode\_no",

 ” framework\_bitcode\_yes” directory appears.

 2. Select the AppCAMOFramework that matches the bitcode setting value.

**Step3. Add AppcamoFramework to your project**

In the project where you want to apply AppCAMO technology, add a framework in the "Show the Project Navigator" screen.

 1. Select your project name with folder icon from the

 Show the Project Navigator list.

 2. Click the right mouse.

 3. Select "Add Files to "Your project name"..."

 4. Add AppcamoFramework.xcFramework.

          check "Copy items if needed" in the Destination item.

 5. In your project->TARGETS->General->

 **Frameworks, Libraries, and Embedded Content**,

          Set the Embedded setting of the added

 AppcamoFramework.xcFramework to "**Embedded&Sign**"

          Then, save the changes to the Xcode settings.

**Step4-1. Modify AppDelegate Source - In case of swift language**

 1. Select **AppDelegate source** file.

 Add code to be added by referring to below.

 PreventFrida.setFridaDetected(): frida detection code

 JailBreak.detectJail() : jailbreak detection code

 PreventDebugger.preventDebug() : debugger and

 emulator detection code

|  |
| --- |
|  **1.import AppcamoFramework** 2.Add JailBreak.detectJail(),PreventDebugger.preventDebug()  func application(\_ application: UIApplication,  didFinishLaunchingWithOptions launchOptions:  [UIApplication.LaunchOptionsKey: Any]?) -> Bool  {               ...               PreventFrida.setFridaDetected()  JailBreak.detectJail()                  PreventDebugger.preventDebug()                return true   } |

 Then, save the changes to code.

**Step4-2. Modify AppDelegate Source**

 **- In case of objective-c language**

1. Select **AppDelegate.m** source file.

 Add code to be added by referring to below.

 [PreventFrida setFridaDetected]: frida detection code

 [JailBreak detectJail]: jailbreak detection code

 [PreventDebugger preventDebug]: debugger and

 emulator detection code

|  |
| --- |
|  1.**#import <AppcamoFramework/JailBreak.h>****#import <AppcamoFramework/PreventDebugger.h>** **#import <AppcamoFramework/PreventFrida.h>** 2.Add JailBreak.detectJail(),PreventDebugger.preventDebug()    (BOOL)application:(UIApplication \*)application  didFinishLaunchingWithOptions:(NSDictionary \*)launchOptions {               ... **[PreventFrida setFridaDetected];**  **[JailBreak detectJail];**   **[PreventDebugger preventDebug];**        // Override point for customization after  application launch.               return YES;   } |

 Then, save the changes to code.

**Step5. Build & upload testflight & Test**

 1. Product>build.

 If you get the Build success result, upload it

 to testflight.

 2. upload testflight

 When uploading to testflight, you can proceed with the

 usual method.

        In the Distribute App, **select App Store Connect** from

 the **Select method of distribution** and upload it.

 3. test

 Download from testflight and run the app.

         Check if jailbreak is detected.