4. Add the ‘CardIO’ directory into the project and skip the next step.

5. Simply add pod ‘CardIO’ into the podfile and update.

6. Go to the TARGET -> BUILD SETTINGS and set OTHER LINKER FLAG to “-lc++” and “-ObjC”

7. Add following frameworks as Optional libraries

- Accelerate
- AVFoundation
- AudioToolbox
- CoreMedia
- CoreVideo
- MobileCoreServices o CoreServices

8. Finally, Confirm that following two settings (TARGET->BUILD SETTINGS) are enabled, if not enable them

- Enable Modules (C and Objective-C)

- Link Frameworks Automatically

Using Objective-C into Swift Project

The Card.io framework is written in Objective-C, and our project is written in Swift. These two don't work together without a proper configuration. Any Objective-C library, project or class can be used in a Swift project by setting up a bridging header.

You can create the bridging file by using following steps,

1. Add a new Header file into project, using File – > New -> File -> iOS -> Source -> Header File.

2. Name the class <your-project-name>-Bridging-Header.h and press continue to save the file.

3. Go to Build Settings -> Objective-C Bridging Header and set as '<your-project-name>-Bridging-Header.h'.

4. Add "#import "CardIO.h" in to your bridging header file.

#import "CardIO.h"

```
@import AudioToolbox;

@import AVFoundation;

@import CoreMedia;

@import CoreVideo;

@import CoreServices;
```