

Introducing Core NFC

Session 718

Lawrence Chung, Wireless Technologies

About Near Field Communication

Core NFC

Requirements

Integration details

Example code

About Near Field Communication

Core NFC

Requirements

Integration details

Example code

Near Field Communication (NFC)

Close proximity wireless communication

RF technologies: NFC-A, NFC-B, NFC-F, NFC-V

NFC Data Exchange Format (NDEF)

Tag
Types

Type 1
Topaz

Type 2
MIFARE Ultralight

Type 3
FeliCa

Type 4
DESFire

Type 5

RF

NFC-A

NFC-B

NFC-F

NFC-V

About Near Field Communication

Core NFC

Requirements

Integration details

Example code

NFC Tag Reading

NFC Tag Reading



NFC Tag Reading



NFC Tag Reading



NFC Tag Reading

NDEF formatted tags

Tag types 1 to 5

Available on iPhone 7 and iPhone 7 Plus

About Near Field Communication

Core NFC

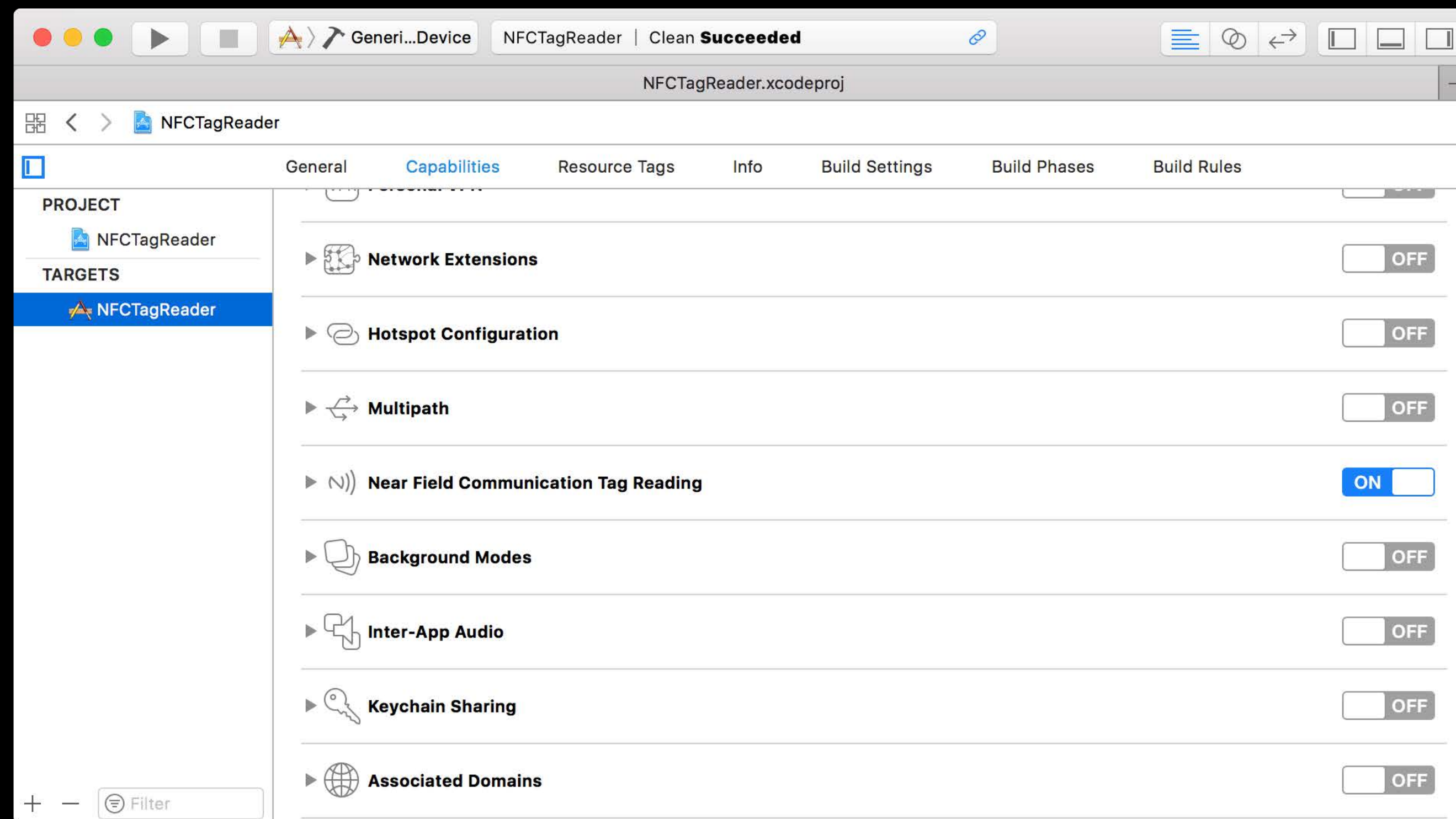
Requirements

Integration details

Example code

NFC Capability

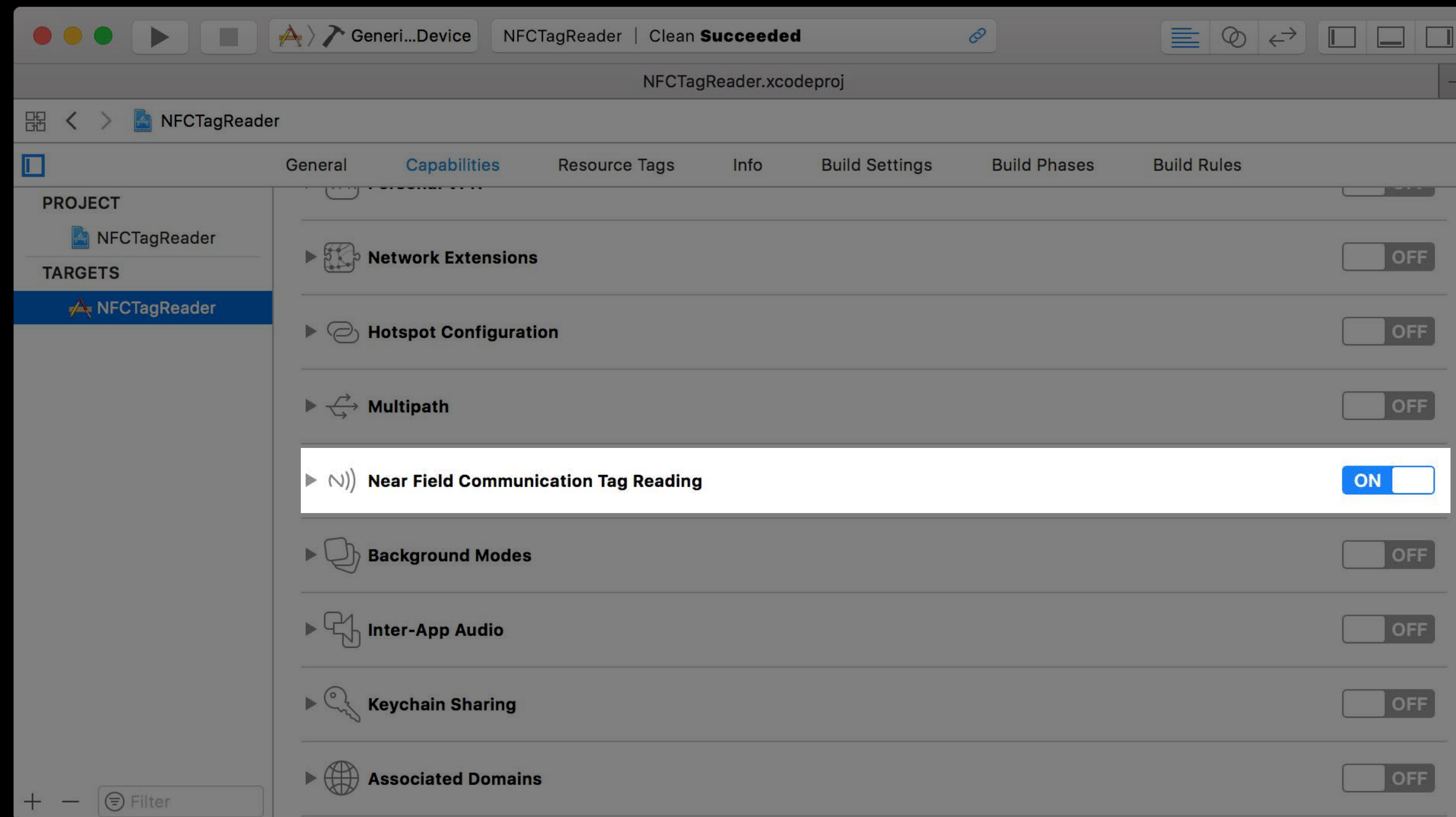
Core NFC requires an entitlement



For initial seed, visit
<https://developer.apple.com/account/ios/certificate>
for manual request

NFC Capability

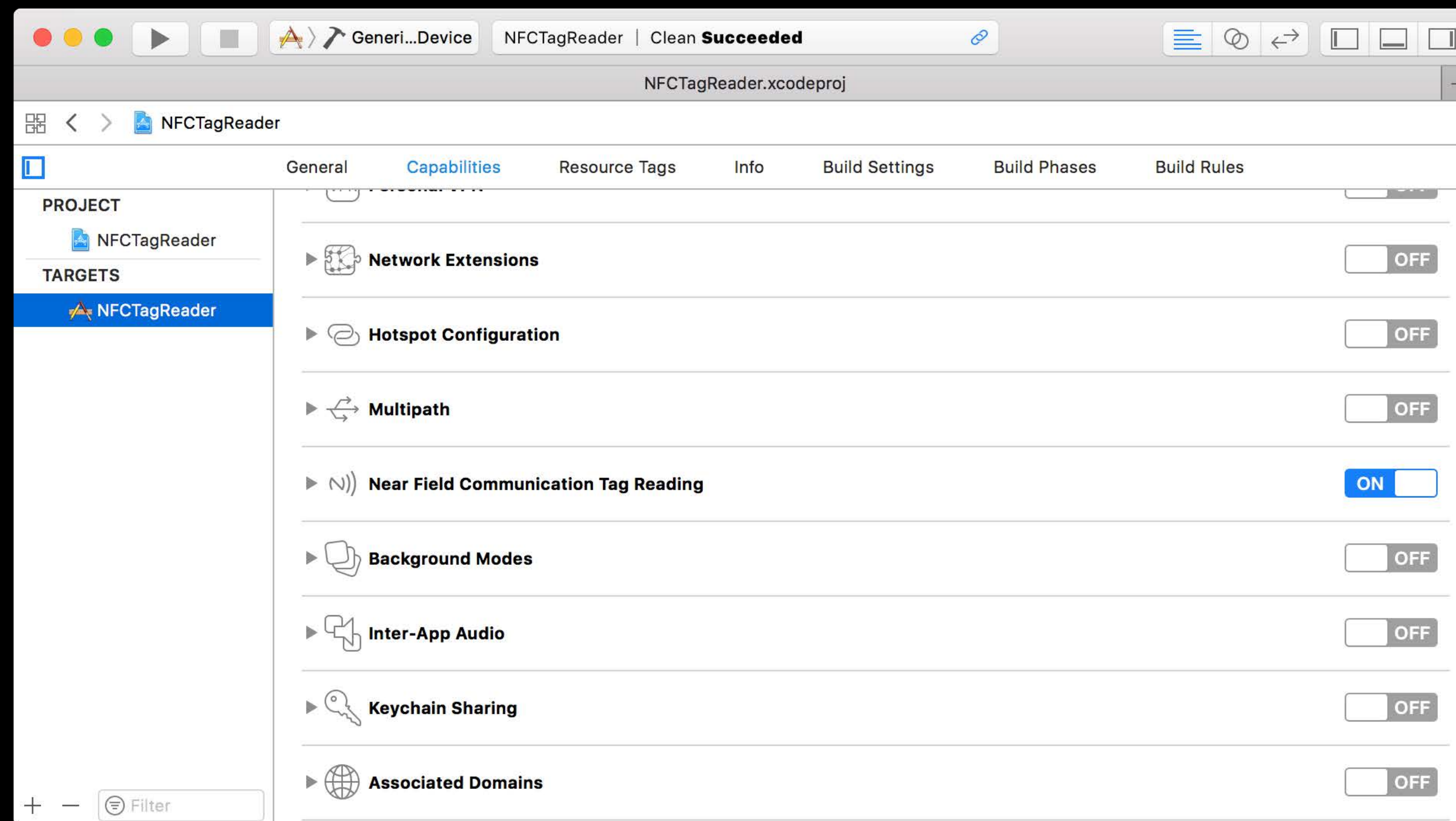
Core NFC requires an entitlement



For initial seed, visit
<https://developer.apple.com/account/ios/certificate>
for manual request

NFC Capability

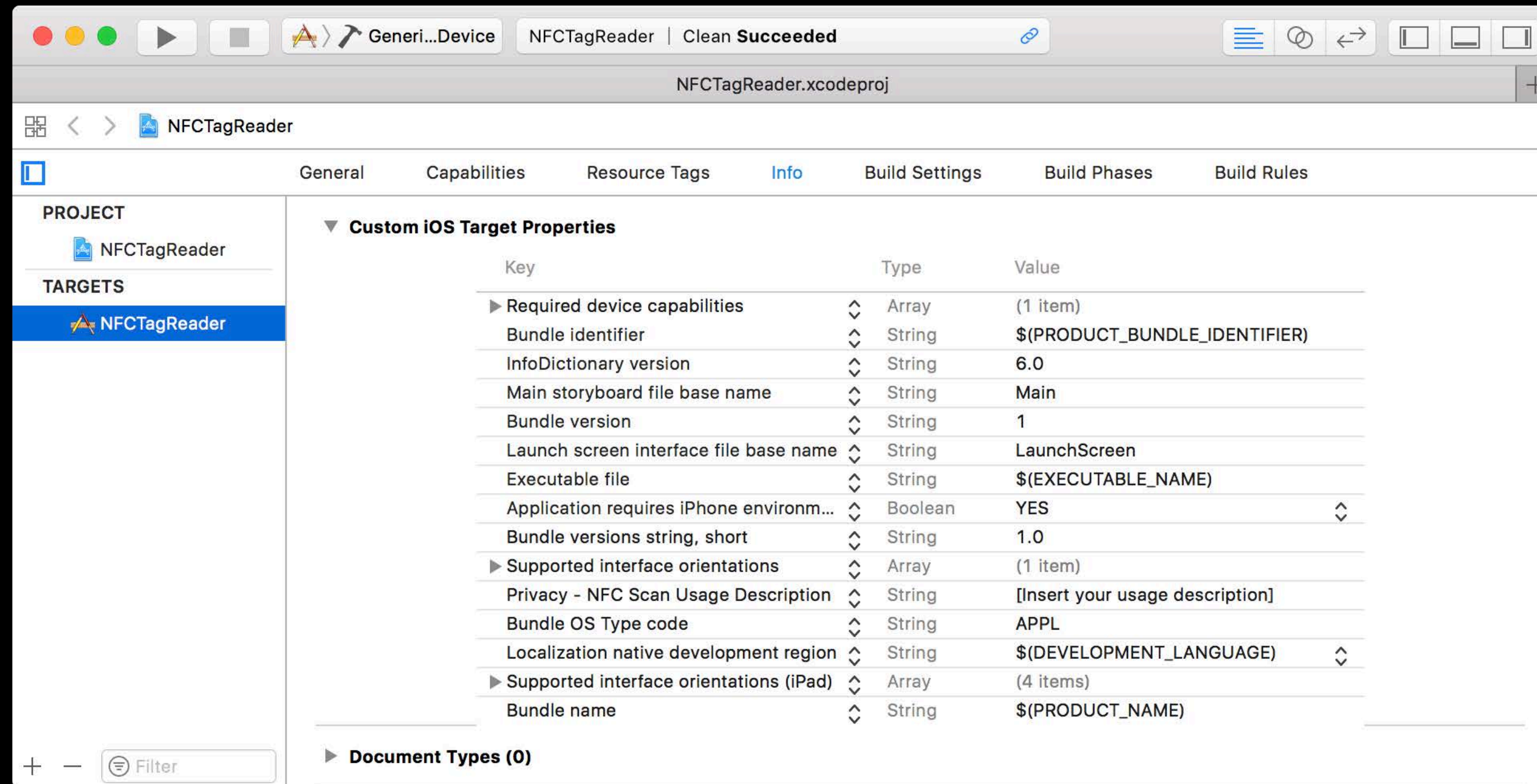
Core NFC requires an entitlement



For initial seed, visit
<https://developer.apple.com/account/ios/certificate>
for manual request

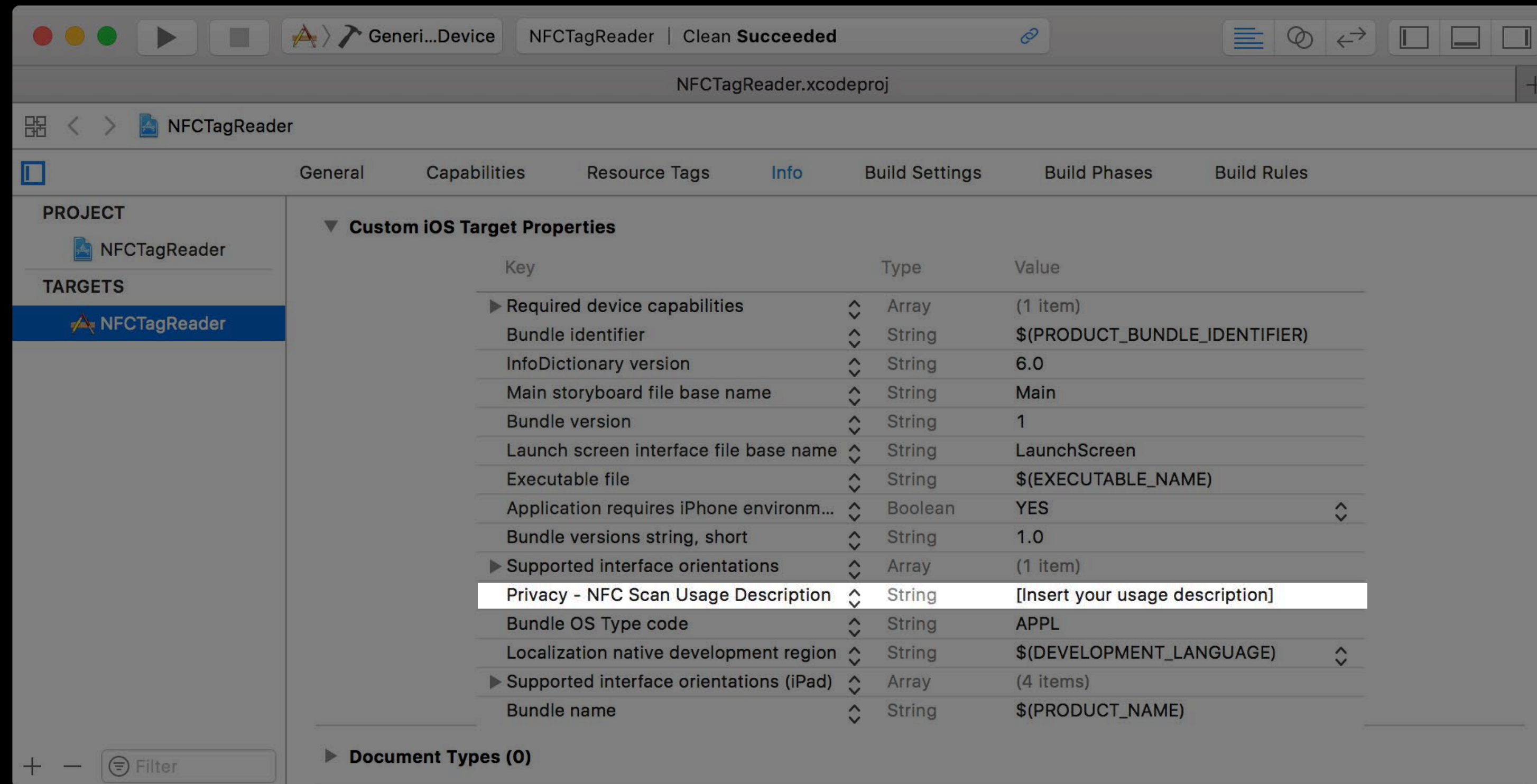
Usage Description String

Add "Privacy - NFC Scan Usage Description" in Info.plist



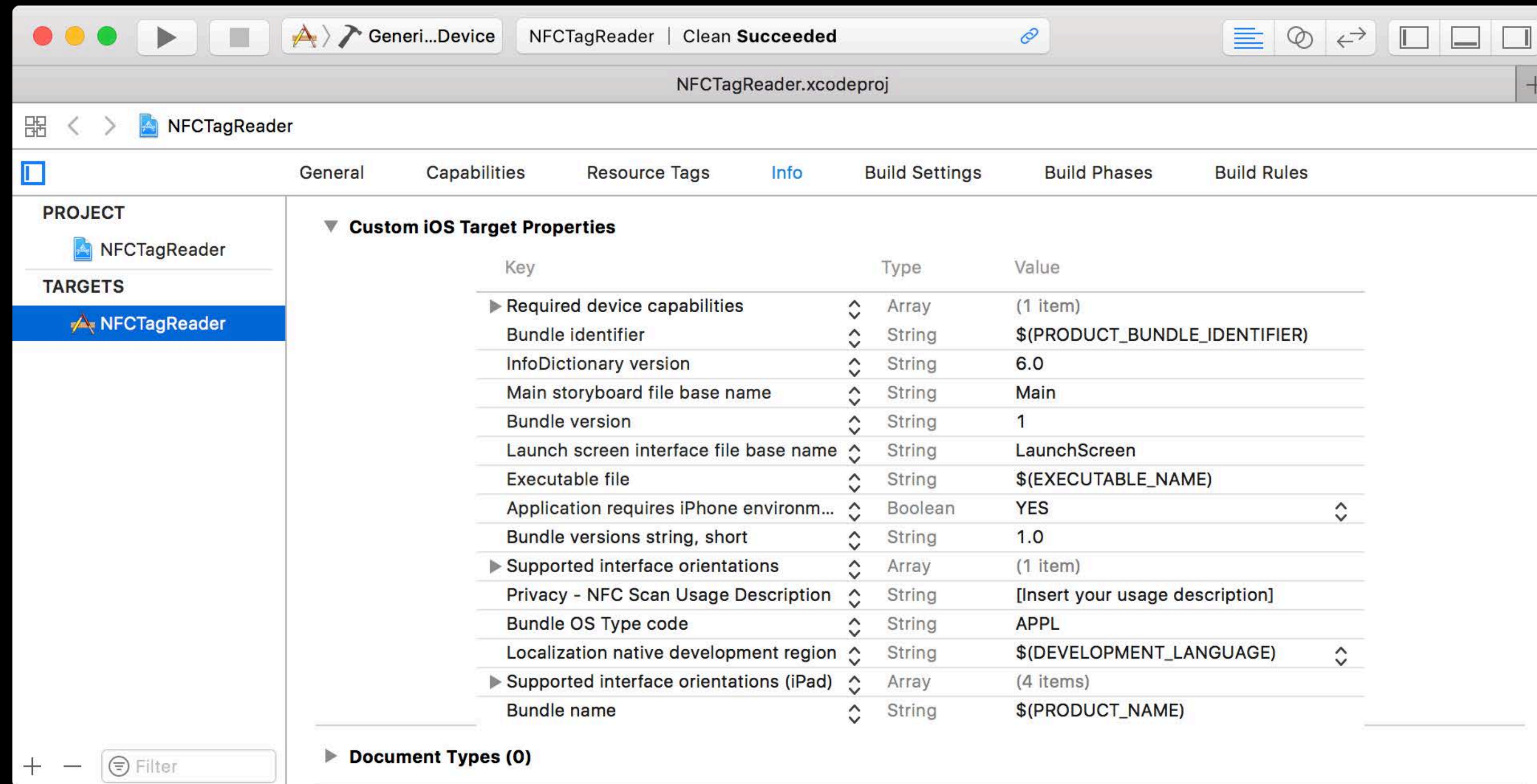
Usage Description String

Add "Privacy - NFC Scan Usage Description" in Info.plist



Usage Description String

Add "Privacy - NFC Scan Usage Description" in Info.plist



About Near Field Communication

Core NFC

Requirements

Integration details

Example code

Important Info

Tag reading details with Core NFC

Important Info

Tag reading details with Core NFC

Session is on demand

Important Info

Tag reading details with Core NFC

Session is on demand

Foreground mode only

Important Info

Tag reading details with Core NFC

Session is on demand

Foreground mode only

Maximum 60 seconds of scanning per session

Important Info

Tag reading details with Core NFC

Session is on demand

Foreground mode only

Maximum 60 seconds of scanning per session

Configurable for single or multi-tag read operation

Important Info

Tag reading details with Core NFC

Session is on demand

Foreground mode only

Maximum 60 seconds of scanning per session

Configurable for single or multi-tag read operation

Usage description will be shown over the top of your application



9:41 AM

100%

Scanned Tags

Scan

Ready to Scan



Hold iPhone near [Insert your usage
description]

Cancel



9:41 AM

100%

Scanned Tags

[Scan](#)

1 Payload >

Ready to Scan



Product Detected



9:41 AM

100%

Scanned Tags

[Scan](#)

1 Payload



About Near Field Communication

Core NFC

Requirements

Integration details

Example code

Using Core NFC

3 easy steps to reading tags

Step 1—Adopt the `NFCNDEFReaderSessionDelegate` protocol

Step 2—Create a `NFCNDEFReaderSession` instance

Step 3—Start session and handle the delegate callbacks

```
//CoreNFC NDEF Reader Session
```

```
import CoreNFC
```

```
class MessagesTableViewController: UITableViewController, NFCNDEFReaderSessionDelegate {
```

```
    // MARK: NFCNDEFReaderSessionDelegate
```

```
    func readerSession(_ session: NFCNDEFReaderSession, didInvalidateWithError error: Error) {
```

```
        // Check invalidation reason from the returned error. Session will be invalidated after  
        the function returns. New session instance is required to restart tag scanning.
```

```
    }
```

```
    func readerSession(_ session: NFCNDEFReaderSession, didDetectNDEFs messages:  
[NFCNDEFMessage]) {
```

```
        // Process read NFCNDEFMessage objects.
```

```
    }
```

```
//CoreNFC NDEF Reader Session
```

```
import CoreNFC
```

```
class MessagesTableViewController: UITableViewController, NFCNDEFReaderSessionDelegate {  
  
    // MARK: NFCNDEFReaderSessionDelegate  
    func readerSession(_ session: NFCNDEFReaderSession, didInvalidateWithError error: Error) {  
        // Check invalidation reason from the returned error. Session will be invalidated after  
        the function returns. New session instance is required to restart tag scanning.  
    }  
  
    func readerSession(_ session: NFCNDEFReaderSession, didDetectNDEFs messages:  
[NFCNDEFMessage]) {  
        // Process read NFCNDEFMessage objects.  
    }  
}
```



```
//CoreNFC NDEF Reader Session
```

```
import CoreNFC
```

```
class MessagesTableViewController: UITableViewController, NFCNDEFReaderSessionDelegate {
```

```
    // MARK: NFCNDEFReaderSessionDelegate
```

```
    func readerSession(_ session: NFCNDEFReaderSession, didInvalidateWithError error: Error) {
```

```
        // Check invalidation reason from the returned error. Session will be invalidated after  
        the function returns. New session instance is required to restart tag scanning.
```

```
    }
```

```
    func readerSession(_ session: NFCNDEFReaderSession, didDetectNDEFs messages:  
[NFCNDEFMessage]) {
```

```
        // Process read NFCNDEFMessage objects.
```

```
    }
```

```
//CoreNFC NDEF Reader Session
```

```
import CoreNFC
```

```
class MessagesTableViewController: UITableViewController, NFCNDEFReaderSessionDelegate {
```

```
    // MARK: NFCNDEFReaderSessionDelegate
```

```
    func readerSession(_ session: NFCNDEFReaderSession, didInvalidateWithError error: Error) {
```

```
        // Check invalidation reason from the returned error. Session will be invalidated after  
        the function returns. New session instance is required to restart tag scanning.
```

```
    }
```

```
    func readerSession(_ session: NFCNDEFReaderSession, didDetectNDEFs messages:  
[NFCNDEFMessage]) {
```

```
        // Process read NFCNDEFMessage objects.
```

```
    }
```

```
//CoreNFC NDEF Reader Session
```

```
import CoreNFC
```

```
class MessagesTableViewController: UITableViewController, NFCNDEFReaderSessionDelegate {
```

```
    // MARK: NFCNDEFReaderSessionDelegate
```

```
    func readerSession(_ session: NFCNDEFReaderSession, didInvalidateWithError error: Error) {
```

```
        // Check invalidation reason from the returned error. Session will be invalidated after  
        the function returns. New session instance is required to restart tag scanning.
```

```
    }
```

```
    func readerSession(_ session: NFCNDEFReaderSession, didDetectNDEFs messages:  
[NFCNDEFMessage]) {
```

```
        // Process read NFCNDEFMessage objects.
```

```
    }
```

```
//CoreNFC NDEF Reader Session
```

```
import CoreNFC
```

```
class MessagesTableViewController: UITableViewController, NFCNDEFReaderSessionDelegate {
```

```
    // MARK: NFCNDEFReaderSessionDelegate
```

```
    func readerSession(_ session: NFCNDEFReaderSession, didInvalidateWithError error: Error) {
```

```
        // Check invalidation reason from the returned error. Session will be invalidated after  
        the function returns. New session instance is required to restart tag scanning.
```

```
    }
```

```
    func readerSession(_ session: NFCNDEFReaderSession, didDetectNDEFs messages:
```

```
[NFCNDEFMessage]) {
```

```
        // Process read NFCNDEFMessage objects.
```

```
    }
```

```
...

// MARK: - Actions
@IBAction func beginScanning(_ sender: Any) {
    let session = NFCNDEFReaderSession(delegate: self, queue: nil, invalidatedAfterFirstRead:
true)
    session.begin()
}
}
```

```
...
```

```
// MARK: - Actions
```

```
@IBAction func beginScanning(_ sender: Any) {
```

```
    let session = NFCNDEFReaderSession(delegate: self, queue: nil, invalidatedAfterFirstRead:  
    true)
```

```
    session.begin()
```

```
}
```

```
}
```

```
...  
  
// MARK: - Actions  
@IBAction func beginScanning(_ sender: Any) {  
    let session = NFCNDEFReaderSession(delegate: self, queue: nil, invalidatedAfterFirstRead:  
true)  
    session.begin()  
}  
}
```

Summary

Summary

Enable NFC tag reading capability

Summary

Enable NFC tag reading capability

Add "Privacy - NFC Scan Usage Description" in Info.plist

Summary

Enable NFC tag reading capability

Add "Privacy - NFC Scan Usage Description" in Info.plist

Add Core NFC to your application

Summary

Enable NFC tag reading capability

Add "Privacy - NFC Scan Usage Description" in Info.plist

Add Core NFC to your application

Happy tag reading!!

More Information

<https://developer.apple.com/wwdc17/718>

Related Sessions

Privacy and Your Apps

Executive Ballroom

Tuesday 11:20AM

What's New in Apple Pay and Wallet

Executive Ballroom

Thursday 5:10PM

Labs

Apple Pay and Wallet

Technology Lab J

Friday 9:00AM–12:00PM

