

PlayFab 웹 인증 브로커 샘플

*\*이 샘플은 2018년 7월 Xbox One XDK와 호환됩니다*

# 설명

Xbox One에서 Xbox Live 서비스를 사용하려면 플레이어가 Xbox Live로 인증을 받아야 합니다. 그 외에도 타이틀에 PlayFab 서비스를 사용하려면 Xbox Live 사용자 계정과 연결 가능한 PlayFab 계정을 만들거나 획득해야 합니다. PlayFab 계정을 얻는 방법에는 여러 가지가 있습니다. 가장 간단한 방법은 처음으로 플레이어가 게임에 로그인 할 때 자동으로 PlayFab 계정을 만드는 것입니다. 이 "간단한" 방법은 Xbox One 게임의 일반적인 로그인 절차이며 대부분의 게임에서 권장되는 시나리오입니다. (PlayFab 인증을 처음 사용하는 경우 SimplePlayFabAuth 샘플로 시작하십시오). PlayFabWebAuthBroker 샘플은 다른 인증 공급자로부터 PlayFab 계정을 공유할 수 있게 하려는 게임에 적합한 보다 고급 로그인 흐름을 보여줍니다. 특히이 샘플은 개방형 인증 공급자 (이 경우 Facebook)와 함께 개방 인증 (OAUTH)을 수행하여 PlayFab 계정을 확보하는 방법을 보여줍니다.

# 샘플 빌드

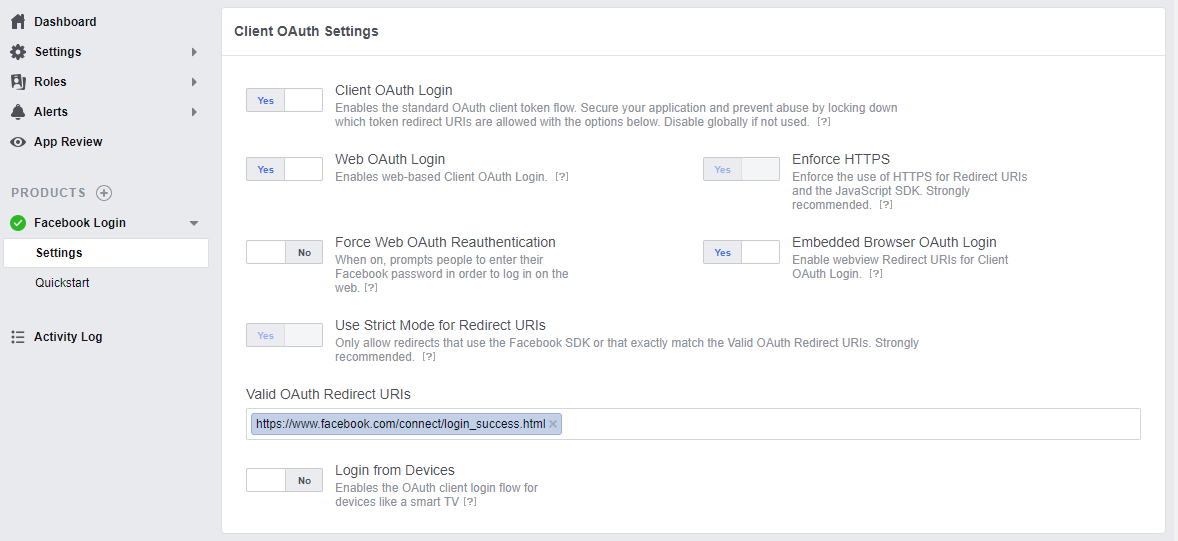
이 샘플은 Visual Studio 2015로 작성되었으며 2018년 7월 Xbox One XDK에서 테스트되었습니다. 이 샘플은 이전 XDK에서 작동할 것입니다

# 샘플용 Facebook 응용 프로그램 구성

이 샘플에서는 Facebook을 예제 OAUTH 공급자로 사용합니다. 이 샘플을 사용하려면 Facebook에 응용 프로그램을 설치해야 합니다. Facebook에서 앱을 설정하는 방법에 대한 자세한 내용은 여기를 참조하십시오. <https://developers.facebook.com/docs/apps>.

## Facebook 로그인 제품

앱을 설정할 때 Facebook 로그인 제품을 추가해야 합니다. 제품을 추가한 후에 유효한 OAuth 리디렉션 URI를 지정해야 합니다. 예를 들면 다음과 같습니다.



## 사용자 테스트

샘플에서 Facebook에 로그인 할 때 사용할 테스트 사용자를 하나 이상 만듭니다. 테스트 사용자 계정은 이미 Facebook에서 앱을 사용할 수 있도록 구성되었으며 샘플을 실험할 때 훨씬 편리합니다.

## APP ID 설정

샘플을 사용하려면 FacebookAuthHelper.h를 편집하고 Facebook APP ID를 제공해야 합니다. Facebook에서 앱의 기본 설정에서 앱 ID를 찾습니다. 그런 다음 FacebookAuthHelper.h에서 "TODO"주석을 찾아 " <put your Facebook App ID here> "앱 ID:



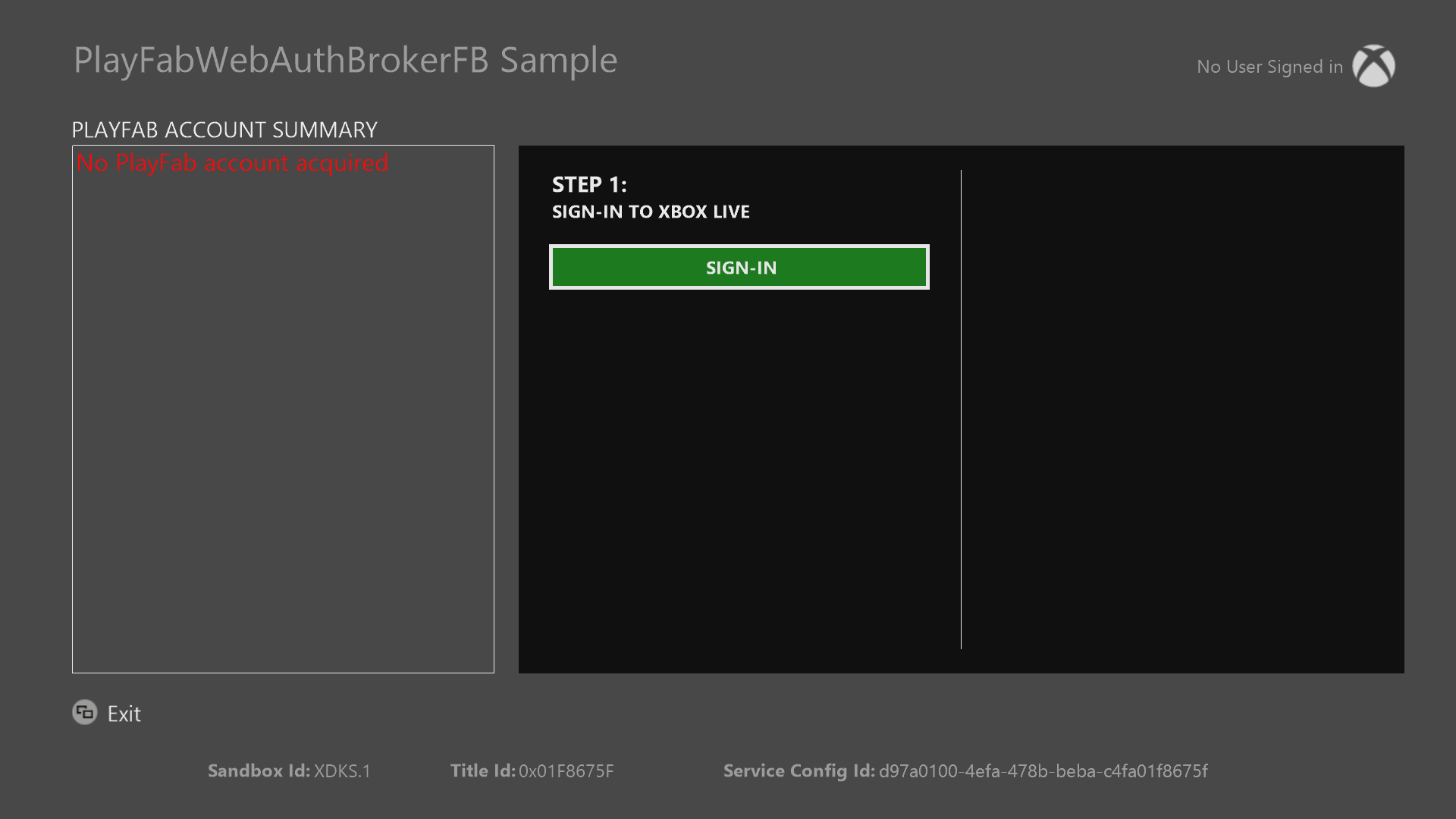
# 샘플 사용하기

일단 Facebook에 모든 것을 설정하고 FacebookAuthHelper.h에 Facebook App ID를 입력하면 샘플을 사용하는 것이 간단합니다. 게임에서 볼 수 있는 것과 유사한 간단한 로그인 흐름이 표시됩니다. 그러나 각 화면에는 일반 사용자에게 표시되지 않는 진단 정보도 표시됩니다. 또한 로그인 절차가 완료되면 계정 요약 정보를 표시하는 패널도 있습니다.

## 로그인 절차

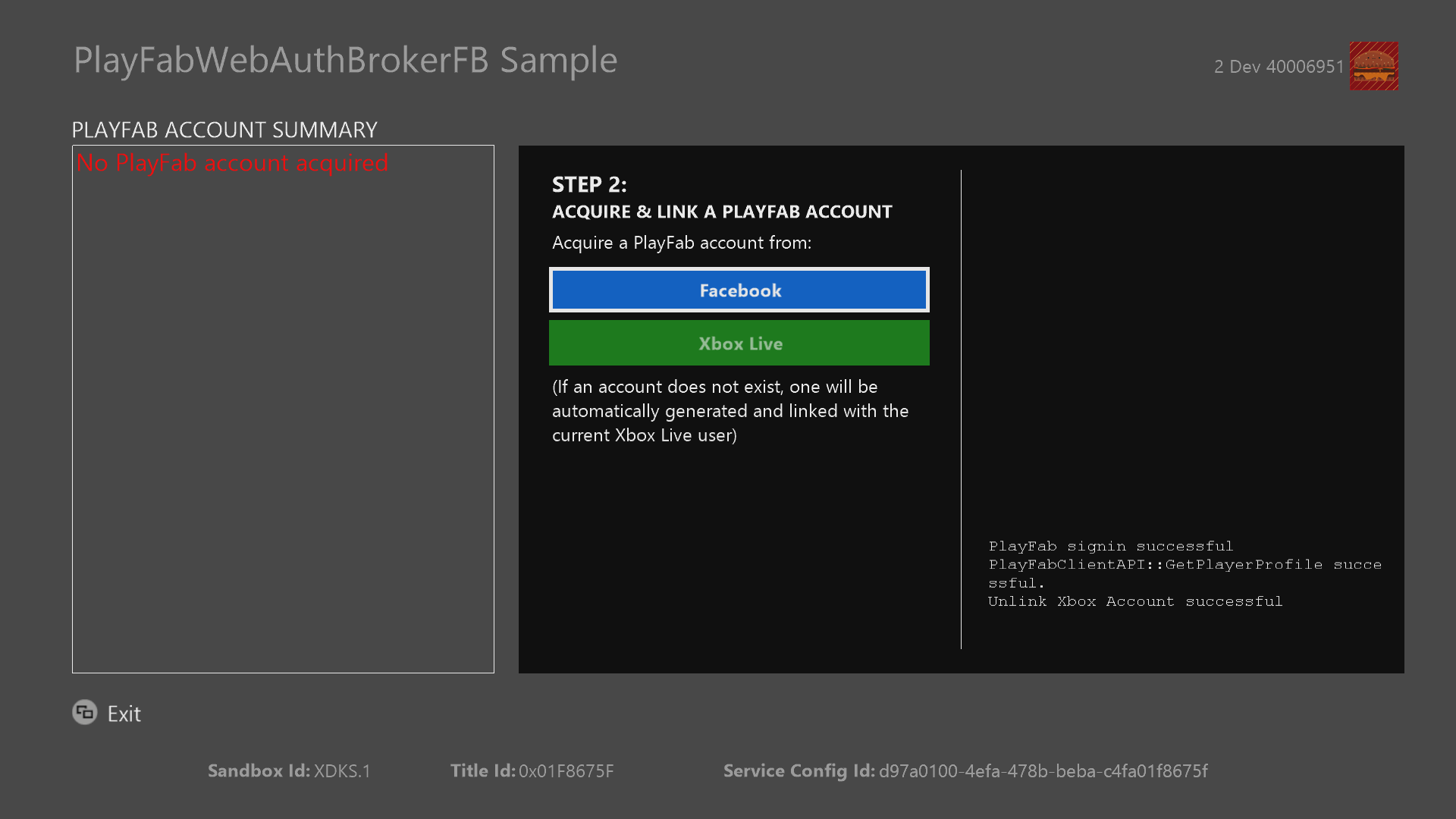
## Xbox Live 로그인

로그인 흐름의 첫 번째 단계는 Xbox Live에 로그인하는 것입니다. Xbox Live 또는 PlayFab에 대한 서비스 요청하기 전에 항상 Xbox Live에 로그인해야 합니다.



## PlayFab 계정 얻기

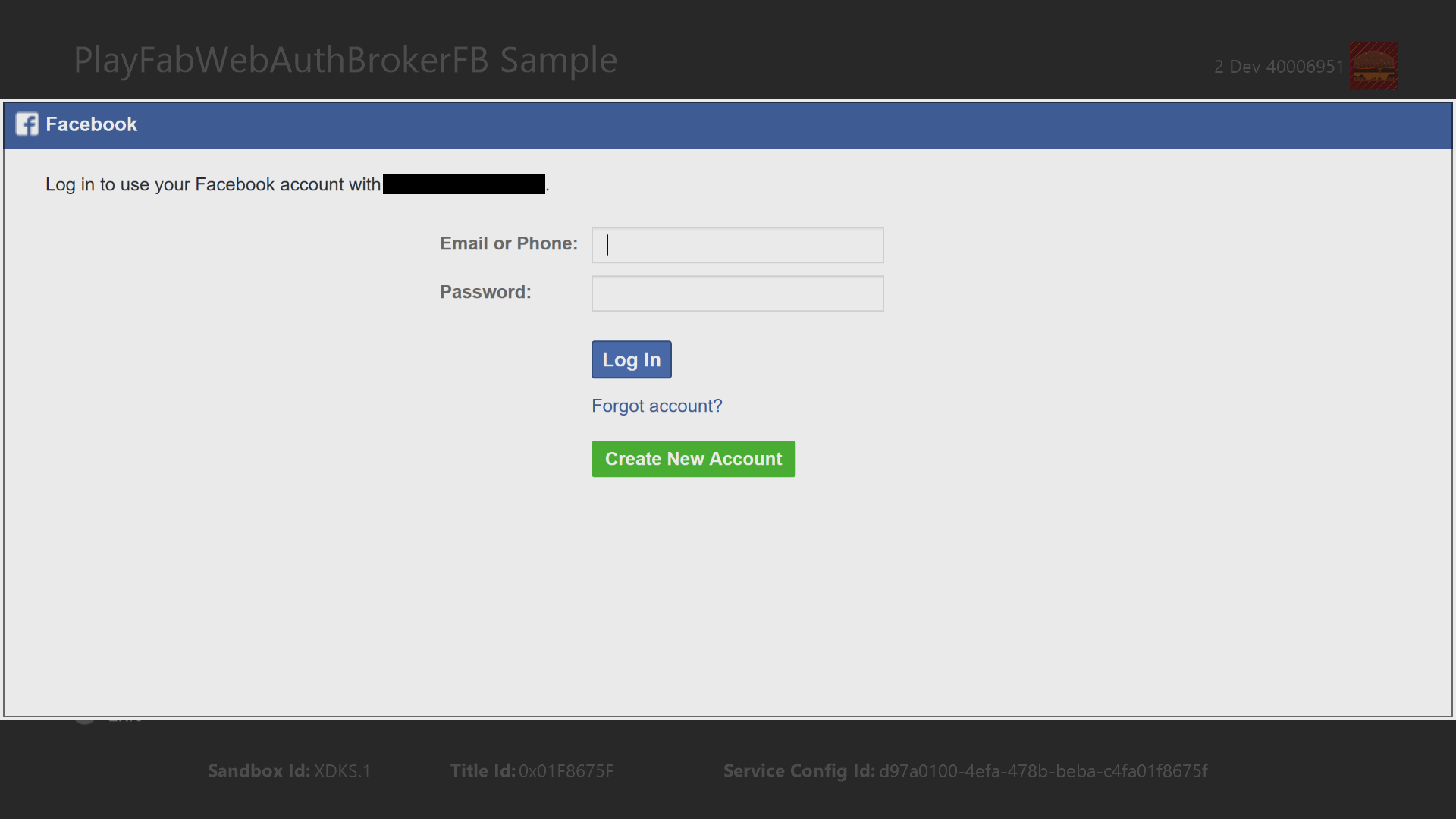
PlayFab 서비스를 사용하려면 PlayFab 계정을 Xbox Live 사용자와 연결해야 합니다. PlayFab 계정을 만드는 방법에는 여러 가지가 있습니다. 로그인 과정의 두 번째 단계에서 사용자에게 두 가지 선택 사항이 제공됩니다. "Xbox Live"선택의 경우 샘플은 Xbox Live 사용자 계정과 연결되는 기본 PlayFab 계정을 간단히 만듭니다. "Facebook"선택의 경우이 샘플은 Facebook과 같은 다른 로그인 공급자에 이미 연결된 PlayFab 계정을 획득하는 방법을 보여줍니다. Facebook에서 OAuth를 수행하면 이미 플레이어의 Facebook 계정과 연결된 PlayFab 계정을 발견한 다음, 이 계정을 플레이어의 Xbox Live 계정과 연결할 수 있습니다.



PlayFab 로그인 옵션에 대한 자세한 내용은에서 PlayFab 설명서를 참조하십시오. [로그인 기본 사항 및 모범 사례](https://api.playfab.com/docs/tutorials/landing-players/best-login)

## Facebook OAuth UI

Facebook에서 인증을 선택하면 샘플에서는 Windows::Xbox::UI::SystemUI::AuthenticateAsync를 호출합니다. 이 API 호출은 다음 웹 기반 시스템 UI를 보여줍니다:



여기에서 Facebook App 용으로 만든 테스트 사용자 중 한 명에게 인증 자격 증명을 제공합니다. 이 UI는 제목이 아닌 시스템에서 렌더링됩니다. 다른 시스템 UI와 마찬가지로이 창은 시스템 키보드 사용을 지원하지만 테스트 목적으로 실제 키보드를 사용하는 것이 더 편리합니다.

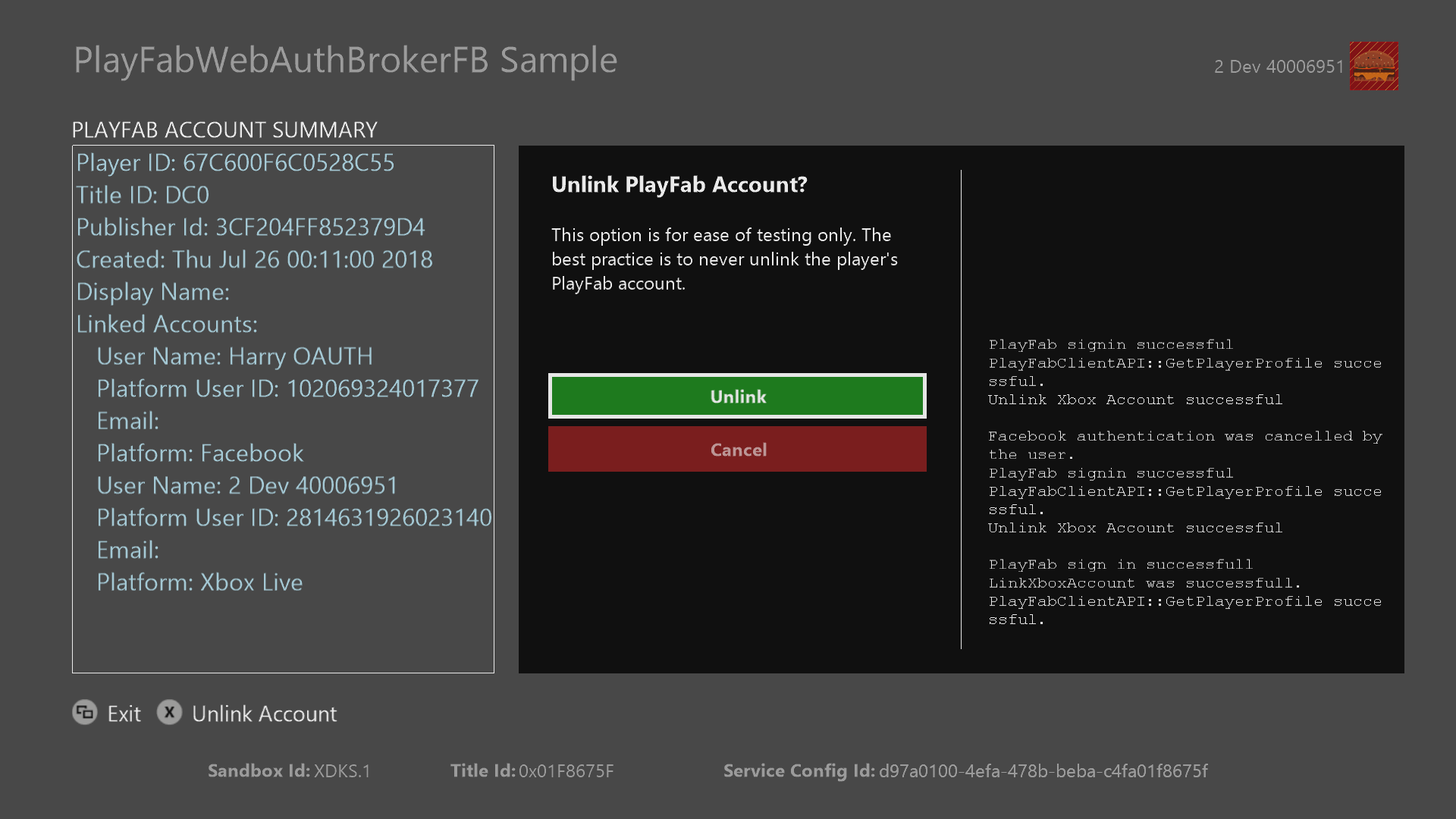
## 로그인 완료

로그인 절차가 완료되면 최종 화면 (3 단계)이 표시됩니다. 이것은 게임이 Xbox Live 및 PlayFab 서비스 요청을 시작할 준비가 되는 지점입니다. 샘플은 화면의 왼쪽에 PlayFab 계정 요약 정보를 표시합니다. 이 정보는 PlayFabClientAPI::GetPlayerProfile에 대한 전화를 통해 제공됩니다.

## 

## Xbox 라이브 계정에서 PlayFab 계정 연결 해제

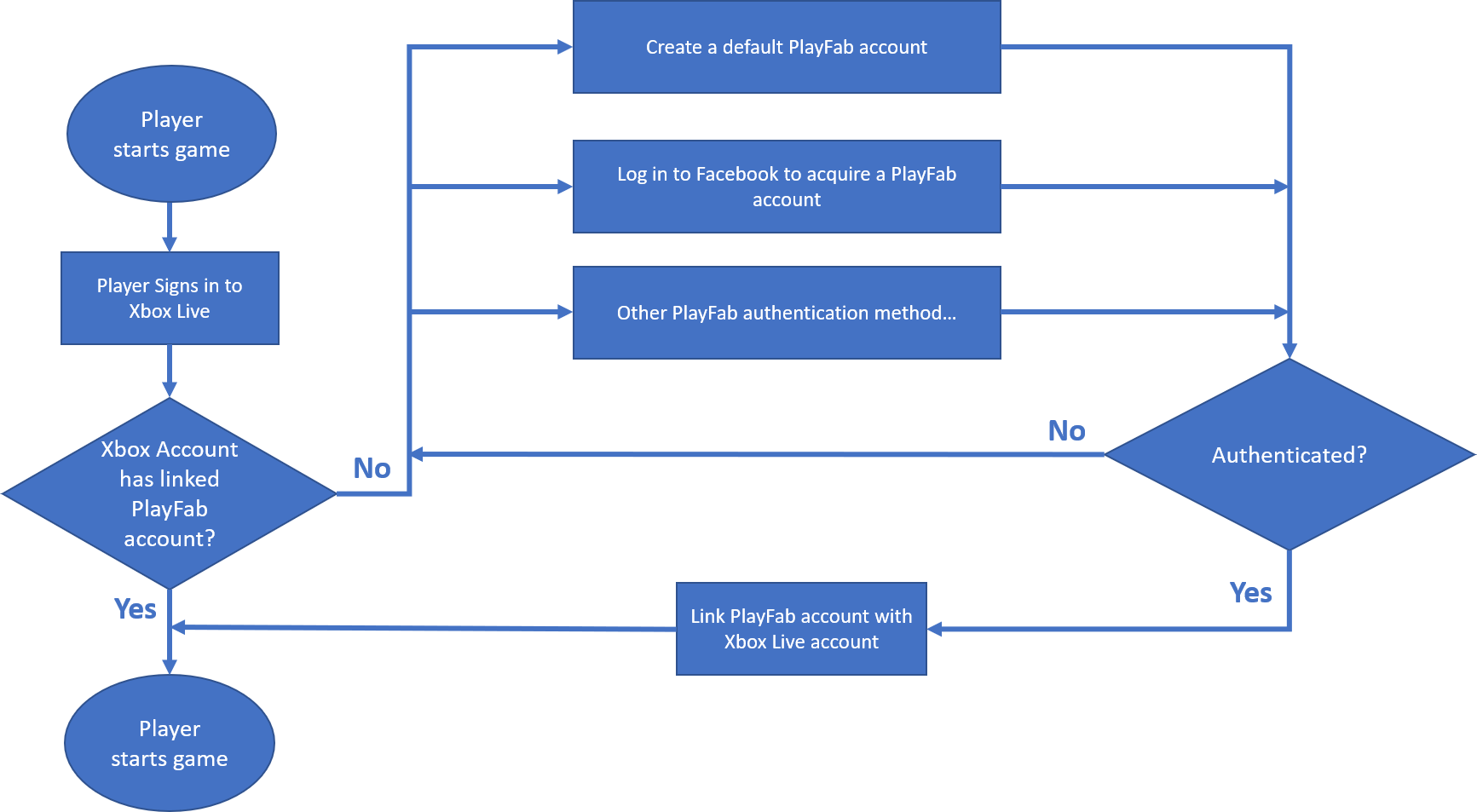
로그인 흐름을 성공적으로 완료하면 샘플에서 PlayFab 계정과 Xbox Live 계정의 연결을 해제하는 옵션을 제공합니다. PlayFab SDK는 API (PlayFabClientAPI::UnlinkXboxAccount)이 목적을 위해, 그러나, 당신은 정상적으로 플레이어의 PlayFab 계정을 연결 해제하지 않을 것입니다. 계정 연결을 해제하면 플레이어 용으로 유지 관리되는 PlayFab 서버측 상태가 손실될 수 있습니다. 또한 동일한 계정 연결을 PlayFab 용 동일한 게시자 ID를 공유하는 여러 제목 간에 매우 잘 공유할 수 있습니다. 따라서 계정 연결 해제의 결과는 플레이어에게 광범위하고 파괴적이며 혼란스러울 수 있습니다. 이러한 이유로 샘플에서는 테스트 목적으로 Unlink 기능만 제공합니다.



# 구현 정보

## 고급 로그인 절차

샘플은 다음과 같은 로그인 흐름을 구현합니다:



이것은 [로그인 기본 사항 및 모범 사례](https://api.playfab.com/docs/tutorials/landing-players/best-login)의 PlayFab 설명서에서 흐름 차트의 단순화된 버전의 순서도입니다. 샘플로 구현된 로그인 흐름은 Xbox One 타이틀에 더 적합합니다. 플레이어가 Xbox Live에 첫 번째 단계로 로그인한다고 항상 가정하기 때문입니다. 논리 로그인 흐름은 PlayFabWebAuthBrokerFB.h/.cpp에서 구현됩니다.

**PlayFab 계정 연결하기**

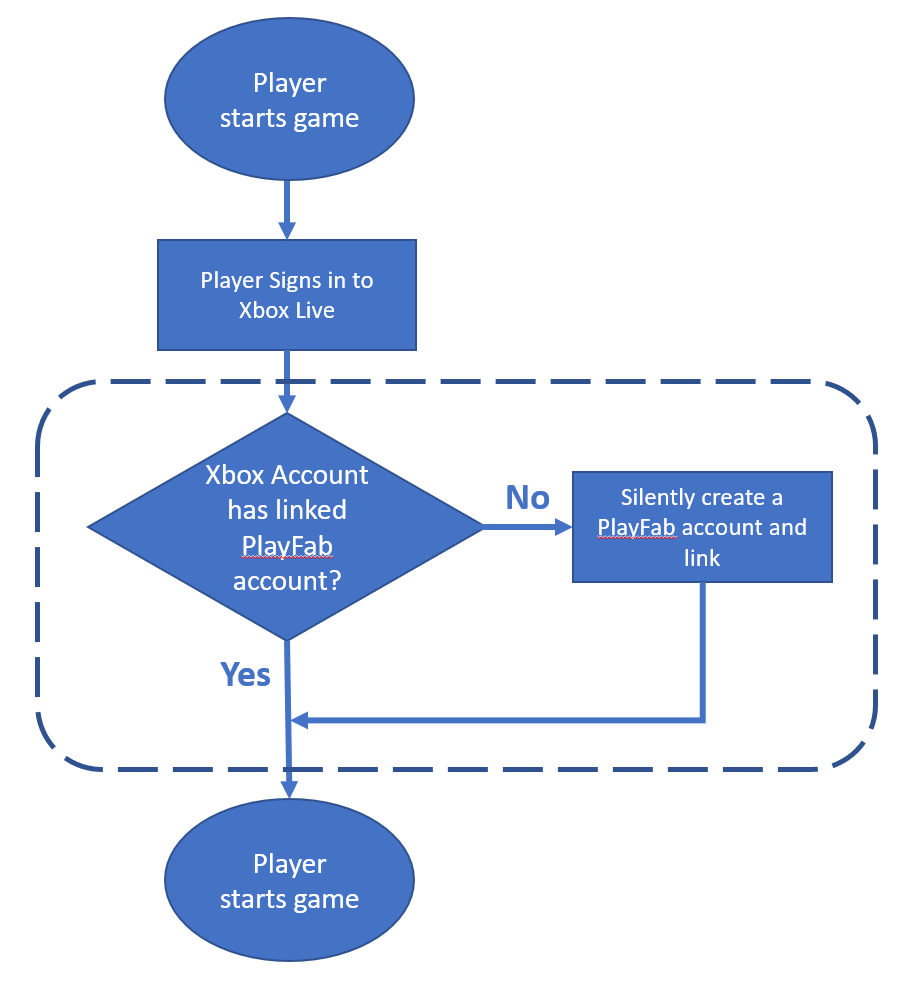
고급 로그인 흐름을 구현할 때 PlayfFab 계정을 Xbox Live 계정에 연결할 때마다 조심해야 합니다. Xbox Live 계정이 이미 PlayFab 계정에 연결되어 있는 경우, 다른 PlayFab 계정을 "강제 연결"하면 원래 계정이 취소됩니다. 플레이어가 원래 PlayFab 계정과 관련된 진행 상태나 다른 계정 상태를 잃어 버리면 플레이어를 혼란스럽고 좌절시킬 수 있습니다. 여기에 표시된 로그인 흐름은 플레이어에게 다른 선택을 제시하기 전에 PlayFab 계정의 존재 여부를 테스트하기 때문에 이 문제를 방지합니다. 구체적으로는 다음과 같은 과정이 진행됩니다:

* PlayFabClientAPI::LoginWithXbox의 오류 처리기에서 샘플은 PlayFabErrorCode::PlayFabErrorAccountNotFound를 확인합니다. 이 오류가 발생하면 샘플의 논리에 PlayFab 계정이 Xbox Live 계정과 이미 연결되어 있음을 알립니다
* PlayFabClientAPI::LinkXboxAccountRequest에서 샘플은 ForceLink 옵션을 ***false***로 설정합니다. 권장 로그인 절차를 따르면 이 옵션을 true로 설정해야 합니다.

가장 좋은 방법은 *ForceLink를 true로 설정할 때 항상 주의하십시오.* 불필요하게 기존 PlayFab 계정을 포기하지 않도록 합니다.

**간단한 로그인 절차**

다음 다이어그램은 최소 로그인 흐름을 보여줍니다.



단순화된 로그인 흐름은 최소화되지만 Xbox Live를 유일한 인증 공급자로 사용하는 대부분의 Xbox One 게임에는 여전히 적합합니다. 점선 안의 모든 것을 수행할 수 있습니다. *아무 말 않고* 플레이어에게 추가 UI 흐름을 제공하지 않아도 됩니다. 특히, LoginWithXboxRequest 구조를 작성할 때 CreateAccount = true를 지정하면 계정이 아직 없는 경우에 새 PlayFab 계정을 만들어야 함을 나타냅니다. 대부분의 게임에서 이 방법이 선호됩니다.

## OAuth 구현

Facebook OAuth 구현은 FacebookAuthHelper라는 독립 실행형 클래스로 제공됩니다. 이것은 "WebAuthBrokerFB"라는 또 다른 샘플의 일부인 "FacebookHelper"의 단순화된 버전입니다. Facebook과 다른 인증 공급자에 대해 OAuth를 수행하려는 경우 예비 사례로 FacebookAuthHelper로 시작하여 적절하게 수정할 수 있습니다.

# 알려진 문제점

## Windows::Xbox::UI::SystemUI::AuthenticateAsync

* 때로는 대화 상자가 응답하지 않고 실제 키보드 입력을 받지 않습니다. 샘플 재시작
* 성공적으로 로그인한 후에는 후속 호출에 대화 상자가 표시되지 않고, 비동기 작업이 성공적으로 완료됩니다. 예를 들어 이전에 성공한 로그인의 캐시된 로그인 결과를 사용할 수 있습니다.
* 성공적으로 로그인 할 때마다 API가 예외를 발생시킵니다. 예외 처리 및 무시할 수 있지만 개발자에게 성가십니다.

# 업데이트 기록

2018년 11월 샘플의 첫 번째 출시

# 개인정보처리방침

샘플을 컴파일하고 실행할 때 샘플 실행 파일의 이름이 Microsoft로 보내져 샘플 사용을 추적 할 수 있습니다. 이 데이터 수집을 거부하려면 Main.cpp에서 "샘플 사용 텔레메트리"라는 코드 블록을 제거하면 됩니다.

Microsoft의 개인 정보 취급 방침에 대한 일반적인 내용은 [Microsoft 개인 정보 취급 방침](https://privacy.microsoft.com/ko-kr/privacystatement/)을 참조하십시오.