SendBird – the messaging app

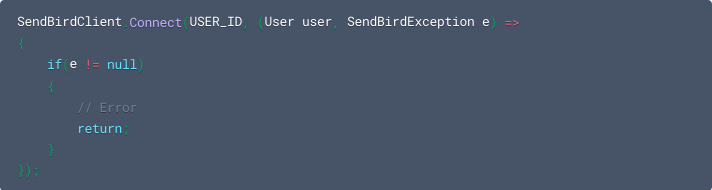
Requirements

* Unity 5.x.x or higher

Steps

* Download Net SDK
* UserInterface
* SendBird unity dispatcher- this needs to be done along with the ‘SendBirdClient.Init’

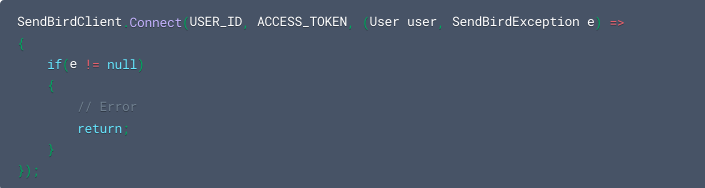
Authentication

* on creating the app, sendBird gives an ‘APP-ID’; we must initialise SendBird using that ‘APP-ID’.
* Requires a ‘USER-ID’ to join the channel.

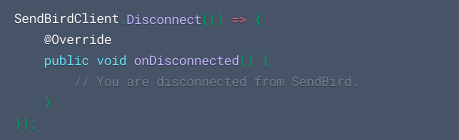
Connecting with UserID and token Access

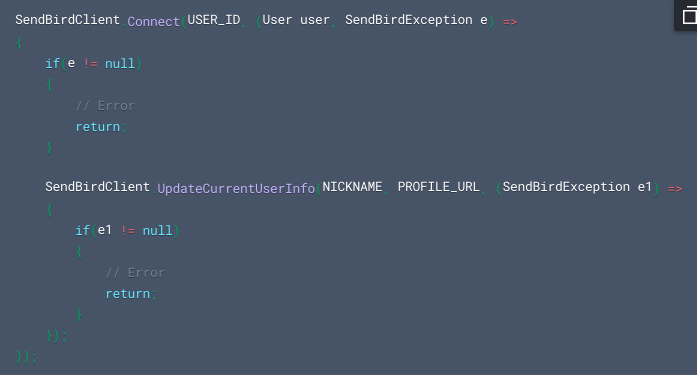
SendBird allows you to create a user with an access token, or you can issue an access for an existing user. The user must provide their users’s token in the login method.

* Create a SendBird user account via the Platform API when your user signs up on your service.
* Save the access token to your secured persistent store.
* Load the access token in your client and pass it to the SendBird login method.
* For security reasons, we recommend that you periodically update your access token by issuing a new token to replace the previous one.

**Note**: You can set restrictions for users without access tokens in your Dashboard settings. These settings can be found under **Security** - **Access Token Policy**.

Disconnecting

Disconnecting removes all registered handlers and callbacks. That is, it removes all Event Handlers added through “SendBirdClient.AddChannelHandler()”

Updating a user profile and profile image

Further instructions follow up from the website

The link of the actual documentation

<https://docs.sendbird.com/unity>