

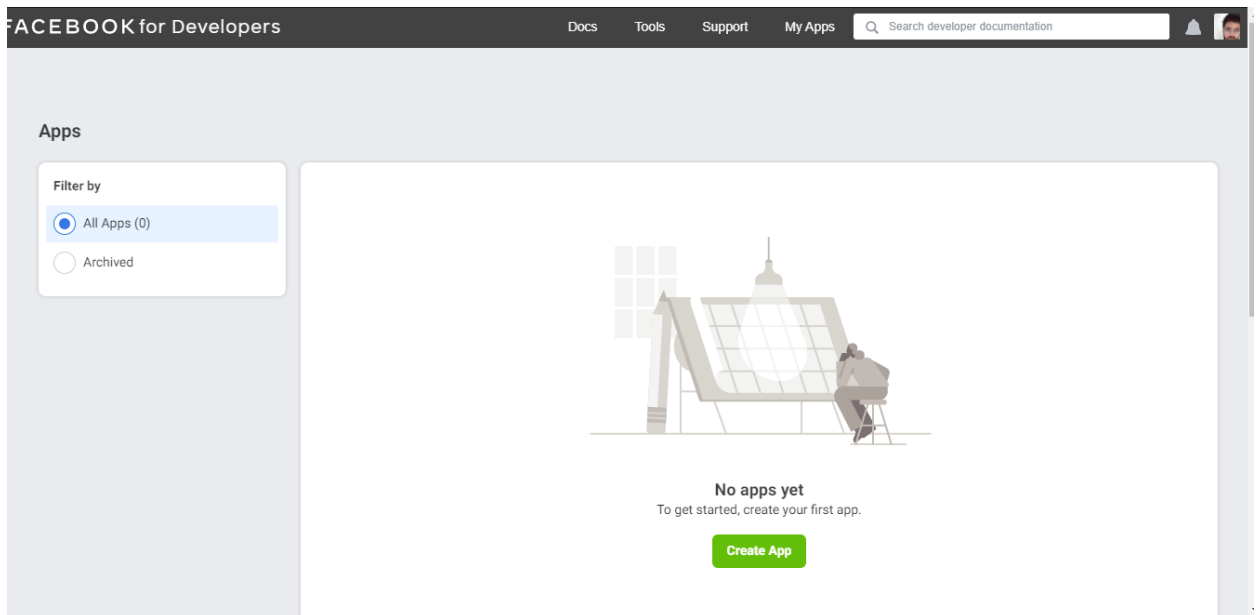
Setting up Facebook Login.

1. Go to Facebook Developers.

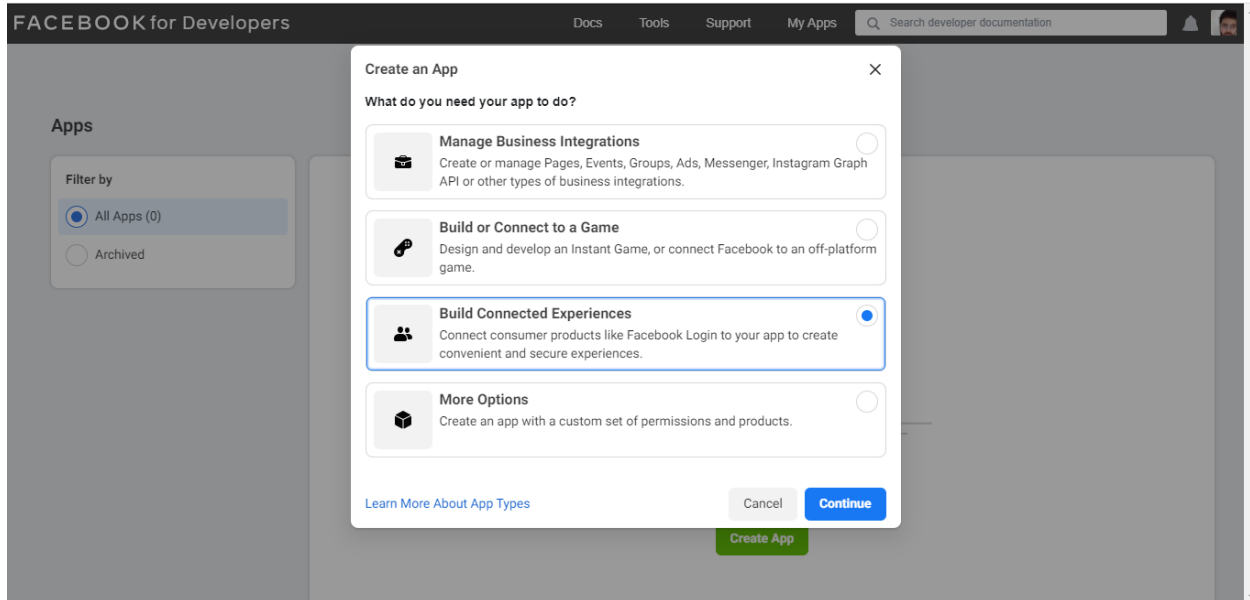
You must login with your facebook account to get access to the Facebook developer panel.

Visit: <https://developers.facebook.com/apps/>

2. Facebook Developers Panel.

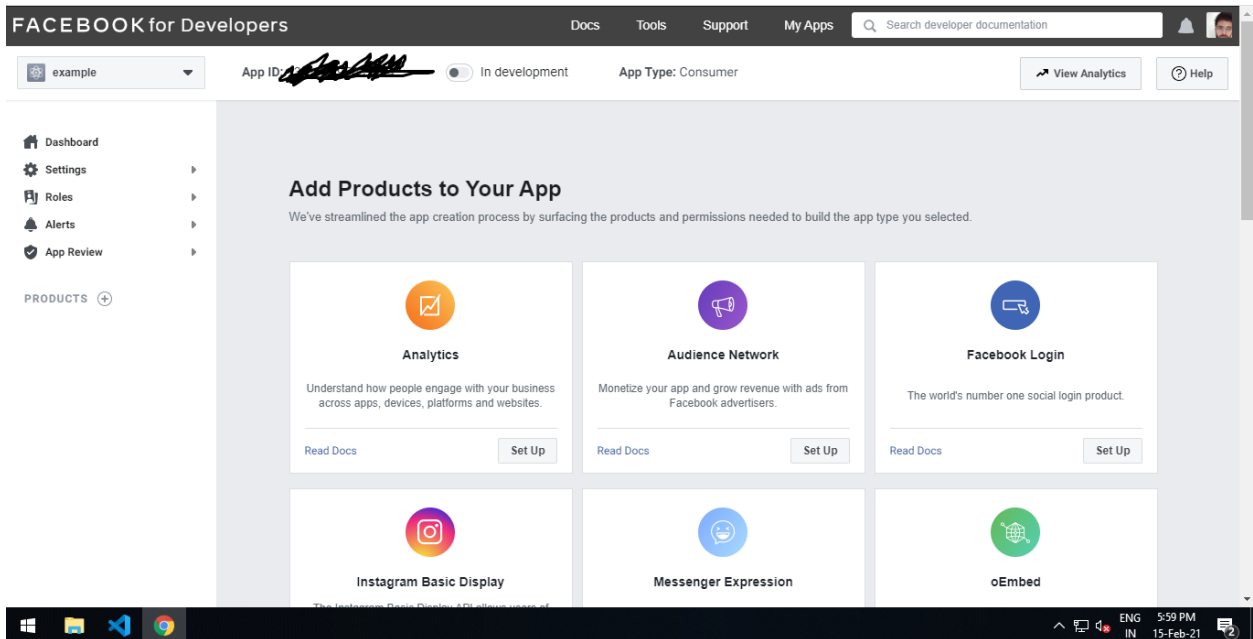


Navigate and Click the create App.



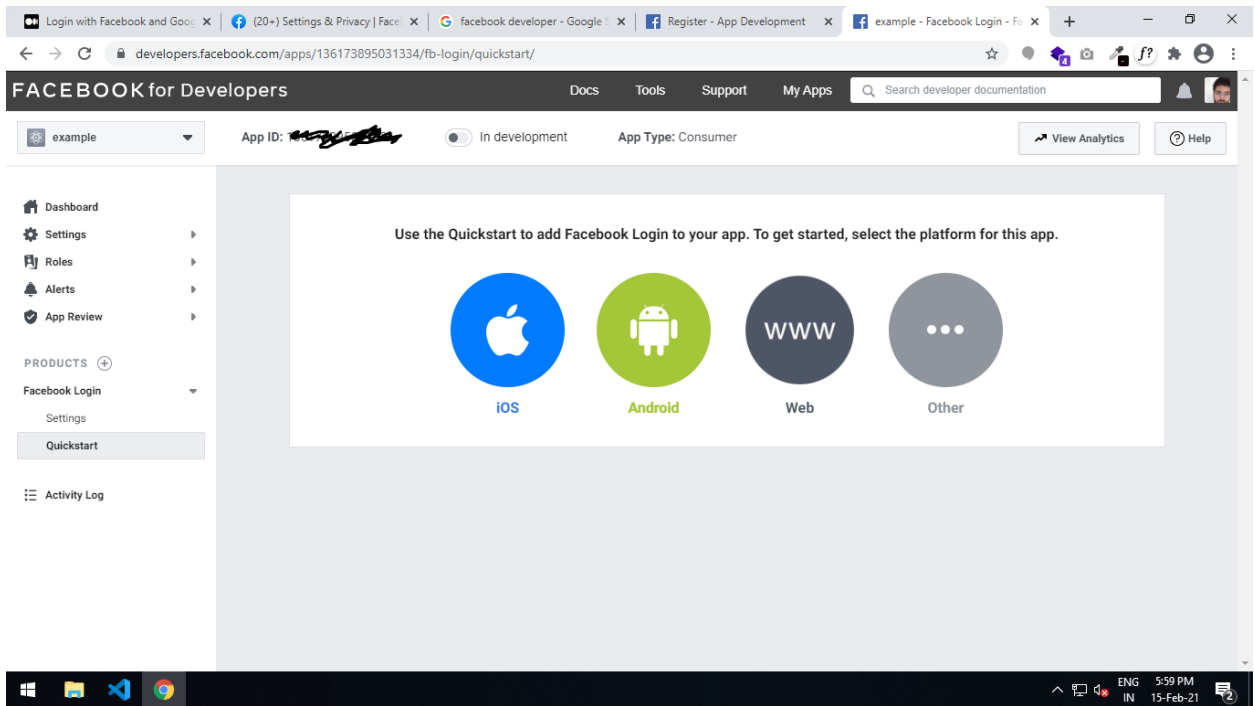
There are various services that are offered when looking for creating an app. Like managing business integration, connecting to a game, **Building connecting experiences like Login that we are going to use.**

3. App Panel and services offered.

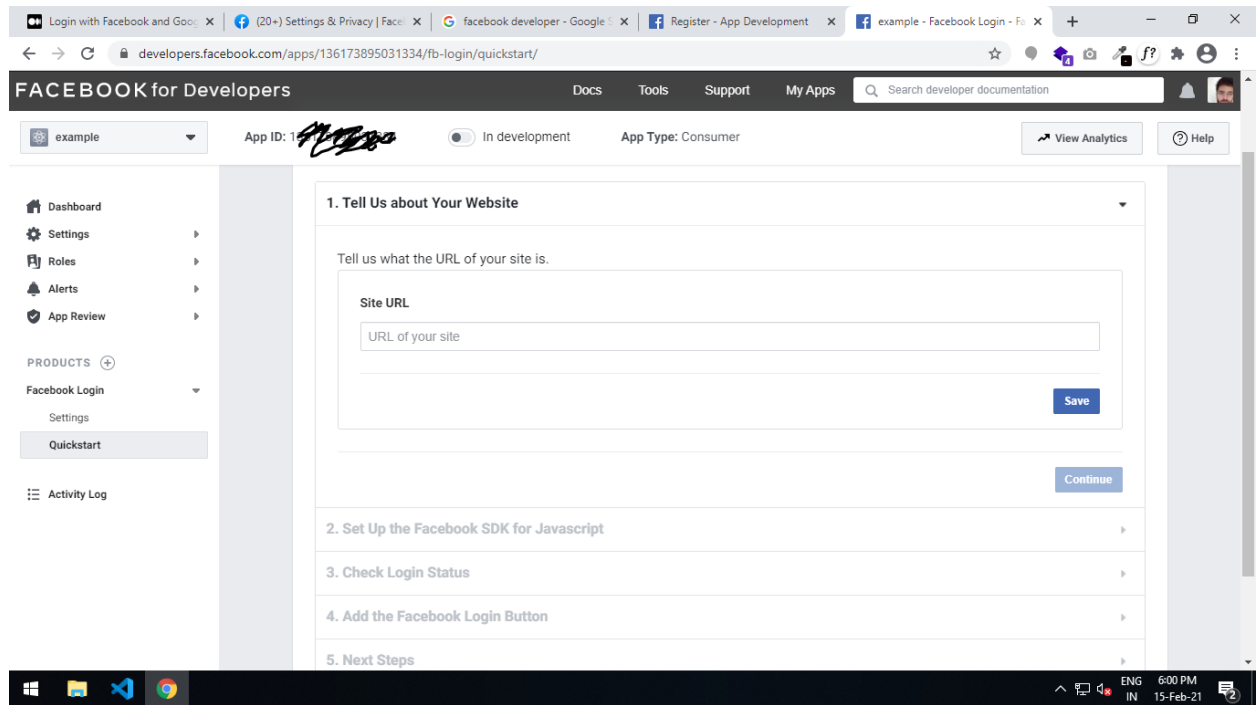


As we can see here Analytics, managing audience, managing facebook login and messenger are offered by the facebook. **Let's click on facebook login.**

4. Login services offered on platforms like web, apple and android platforms.

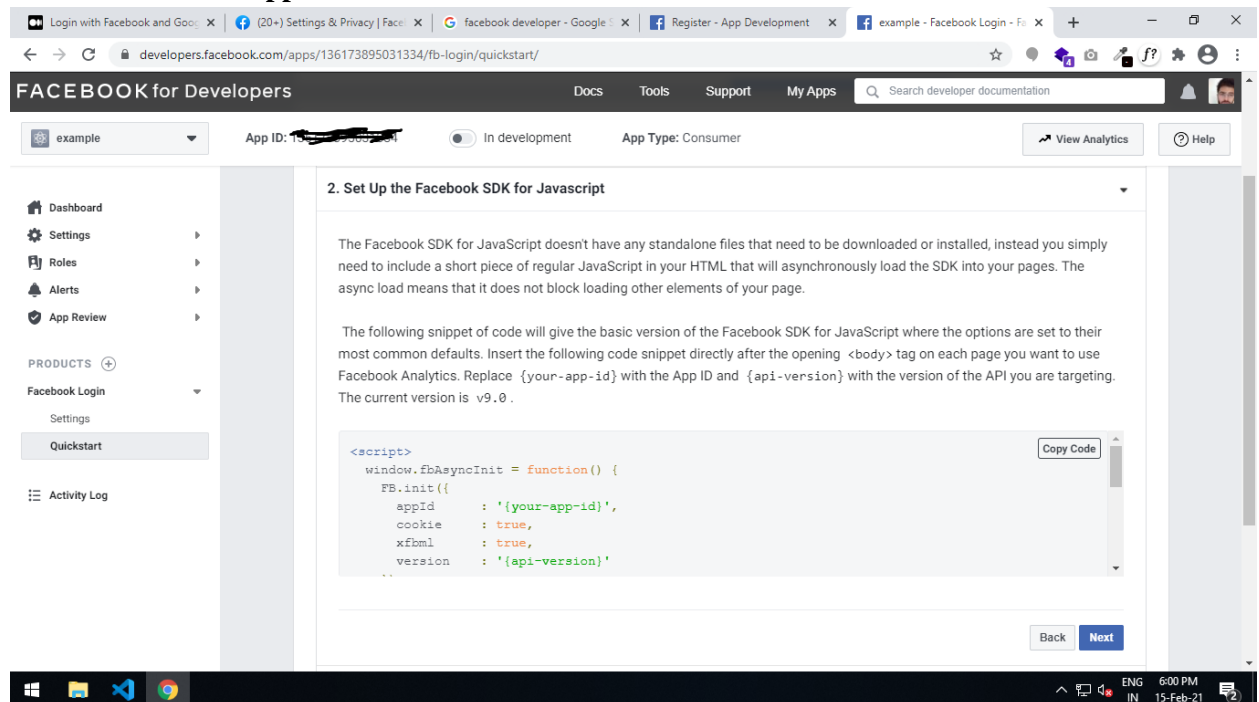


5. For integrating login on website click on WWW and enter your website URL in which you want to implement the login functionality.



6. Setup the SDK. It's a pre-written code already for the developer so that he/she can copy paste and use it directly without going too deep.

Please Note-> The App ID must be confidential and must be hidden from outsider.



7. For Checking if the user is logged In or not on your website the following functions are provided to you are-

The screenshot shows the Facebook for Developers dashboard for an app named 'example'. The left sidebar contains navigation links: Dashboard, Settings, Roles, Alerts, App Review, PRODUCTS (+), Facebook Login (with sub-links for Settings and Quickstart), and Activity Log. The main content area is titled '3. Check Login Status'. It explains that the first step is to check if a user is logged in using the `FB.getLoginStatus()` function. A code snippet is provided:

```
FB.getLoginStatus(function(response) {
  statusChangeCallback(response);
});
```

 Below this, it states that the `response` object contains fields like `status`, `authResponse`, `accessToken`, `expiresIn`, `signedRequest`, and `userID`. A sample `response` object is shown:

```
{
  status: 'connected',
  authResponse: {
    accessToken: '...',
    expiresIn: '...',
    signedRequest: '...',
    userID: '...'
  }
}
```

8. To add & test the login a custom Button Html code is provided. You can write your own custom code.

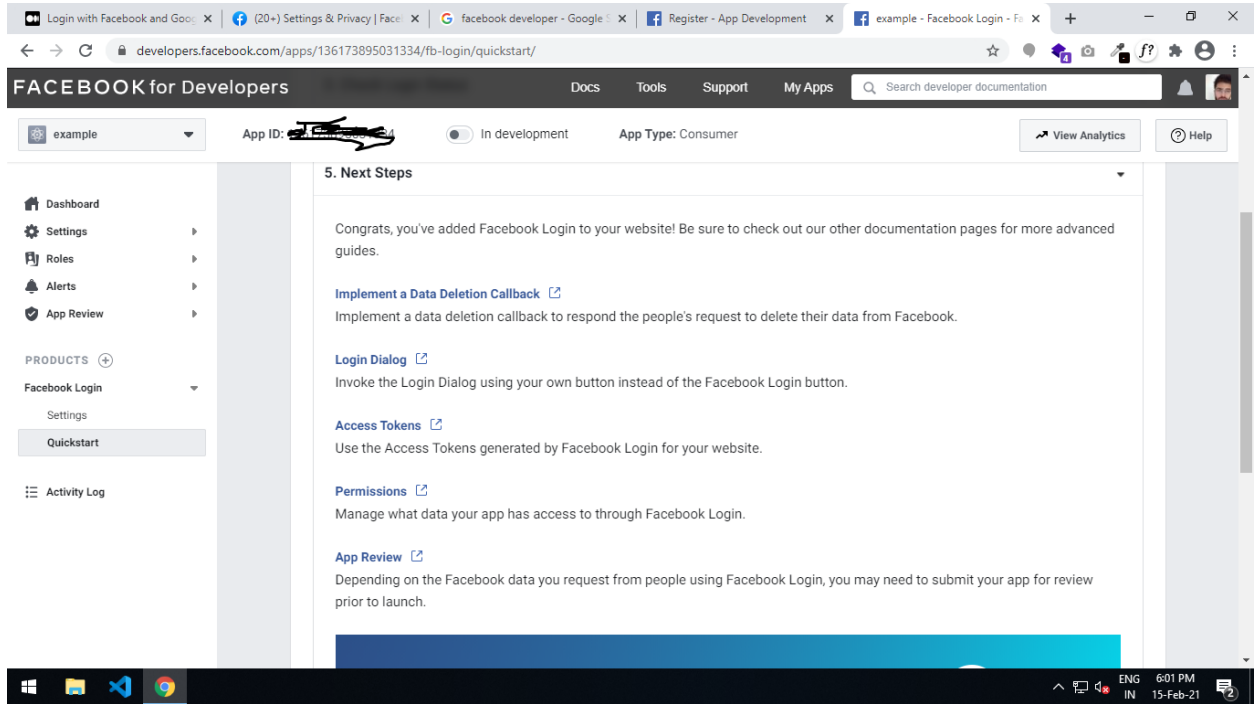
The screenshot shows the Facebook for Developers dashboard for the same app 'example'. The left sidebar is identical to the previous screenshot. The main content area is titled '4. Add the Facebook Login Button'. It explains that including the login button is easy and provides a code snippet for the button:

```
<fb:login-button
  scope="public_profile,email"
  onlogin="checkLoginState();">
</fb:login-button>
```

 It also mentions the `onlogin` attribute and provides a JavaScript callback function:

```
function checkLoginState() {
  FB.getLoginStatus(function(response) {
    statusChangeCallback(response);
  });
}
```

9. So after All these steps! You have successfully implemented a simple Facebook Login Functionality and you are ready to implement the same in your website.



THANK YOU!