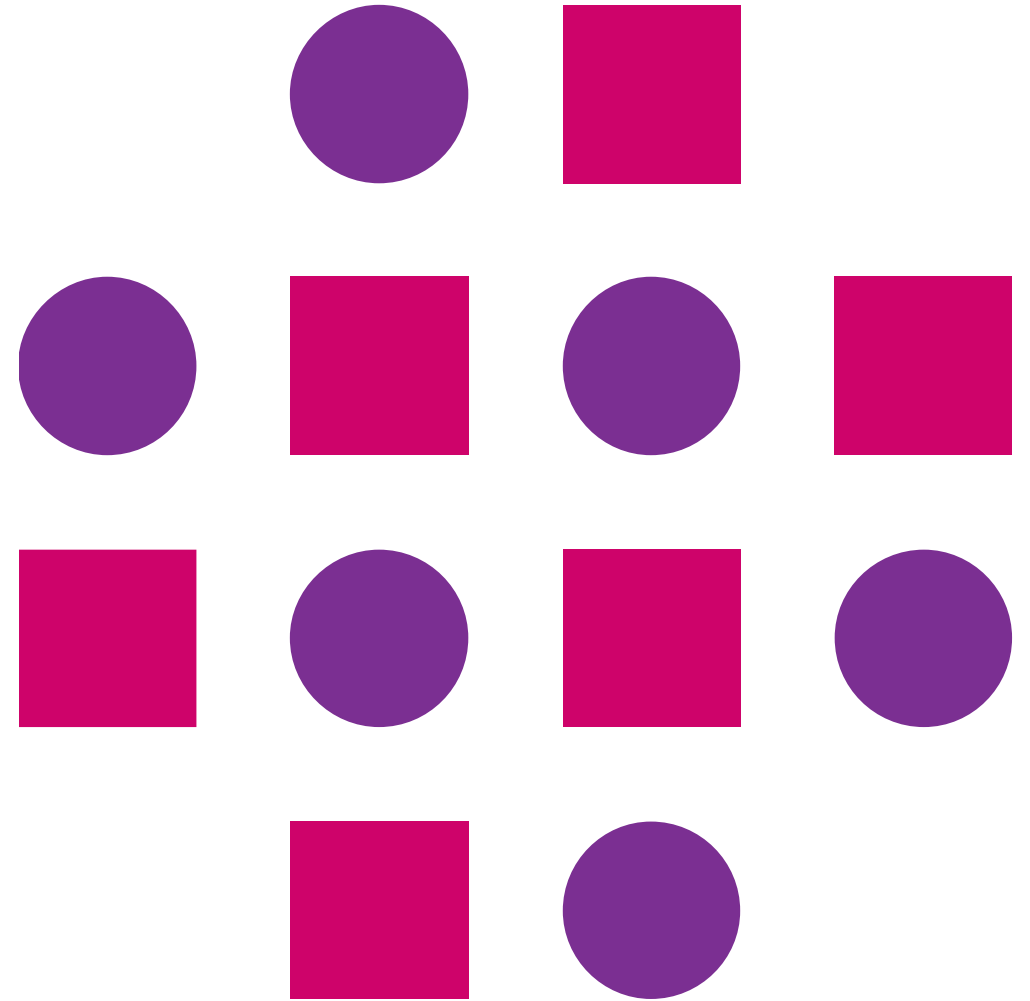


Social-Network Authentication

Angular





WSA

Forward looking IT finishing school

Social Network Authentication

(Login to my Application with your Social Network Account)

Social Network Authentication

- Almost every website which contains a log in option, also contains ways to log in through different social networks, like
 - **Google**
 - **Facebook**
 - **Twitter**
 - **Instagram**
 - **Linked In, etc.**

Google Authentication

(Login to Website through Google Account)

Google Authentication

- We are going to authenticate to Google using Firebase. Follow the steps:

Step 1: Enable the Google authentication in your firebase project.

Step 2: Copy the configuration of your firebase project in environments folder.

Step 3: In app.module.ts add the following imports

```
import { AngularFireAuthModule } from '@angular/fire/auth';
import { AngularFireModule } from '@angular/fire'
import { environment } from 'src/environments/environment';
@NgModule({
  declarations: [
    AppComponent,
    GoogleComponent
  ],
  imports: [
    BrowserModule,
    AngularFireModule.initializeApp(environment.firebase, 'angular-fs'),
    AngularFireAuthModule
  ],
  providers: [],
  bootstrap: [AppComponent]
})
```

Google Authentication

Step 4: Create the service file to send and get response from google server. Write the following code to login

```
import { Injectable } from '@angular/core';
import * as firebase from 'firebase';
import { AngularFireAuth } from '@angular/fire/auth';
import { Observable } from 'rxjs';

@Injectable({
  providedIn: 'root'
})
export class GoogleAuthService {
  user$: Observable<firebase.User>;
  constructor(private afAuth:AngularFireAuth) {
    this.user$ = afAuth.authState;
  }
  login() {
    return this.afAuth.auth.signInWithRedirect(new firebase.auth.GoogleAuthProvider());
  }
}
```

Google Authentication

Step 5: Inject the service in the component and call the login method and get the user details.

```
import * as firebase from 'firebase';
import { AngularFireAuth } from '@angular/fire/auth';
import { GoogleAuthService } from '../services/google-auth.service';

@Component({...})
export class GoogleComponent implements OnInit {
  user;
  constructor(private gService:GoogleAuthService) {
  }
  ngOnInit() {
    this.gService.user$.subscribe(user=> {
      this.user = user;
    });
  }

  loginGoogle() {
    this.gService.login()
  }
}
```


Google Authentication

Logout from Google:

In service write this code:

```
logout() {  
    this.afAuth.auth.signOut();  
}
```

In component call the method:

```
logout() {  
    this.gService.logout();  
}
```

Enabling and disabling the button:

```
<button class="btn btn-primary" (click)="loginGoogle()" *ngIf="!user">  
    Login With Google  
</button>  
<button class="btn btn-danger" (click)="logout()" *ngIf="user">  
    Logout  
</button>
```




Forward looking IT finishing school

Facebook Authentication

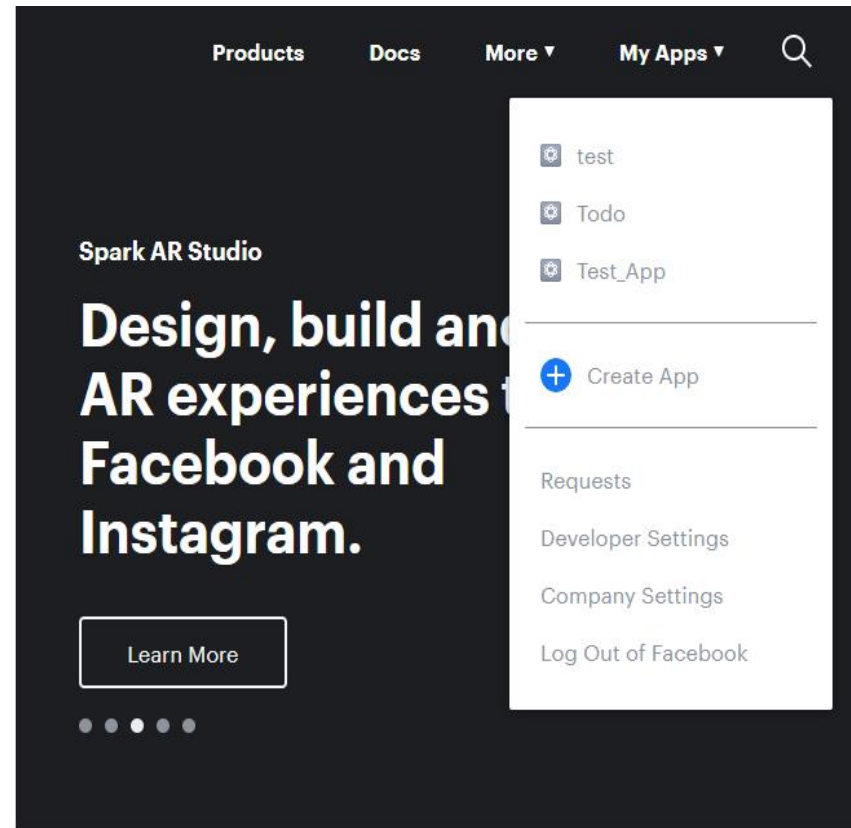
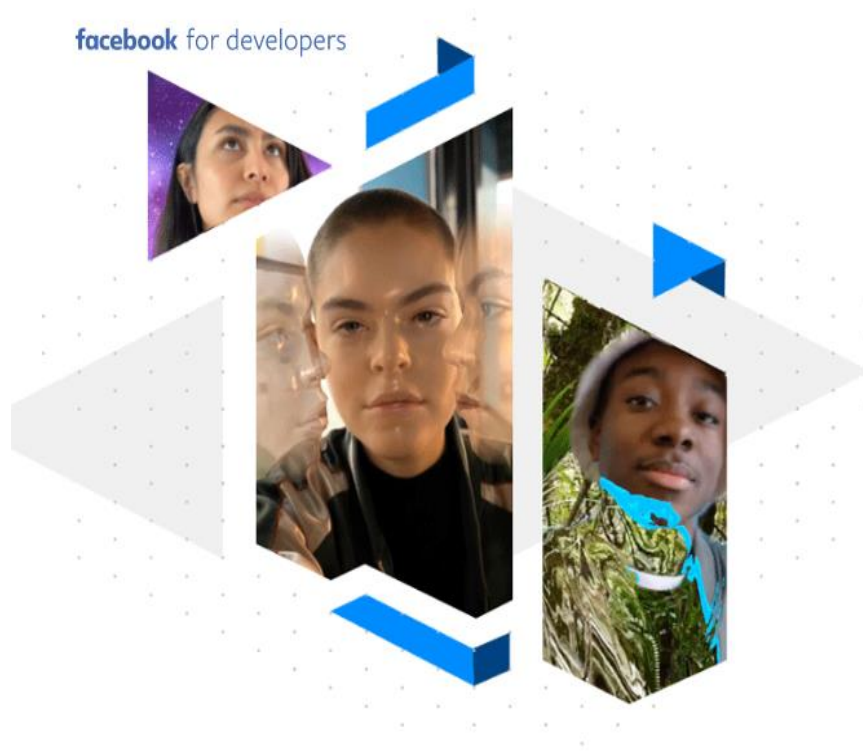
(Login to Website through Facebook Account)

Facebook Authentication

- We are going to authenticate to Facebook using Firebase. Follow the steps:

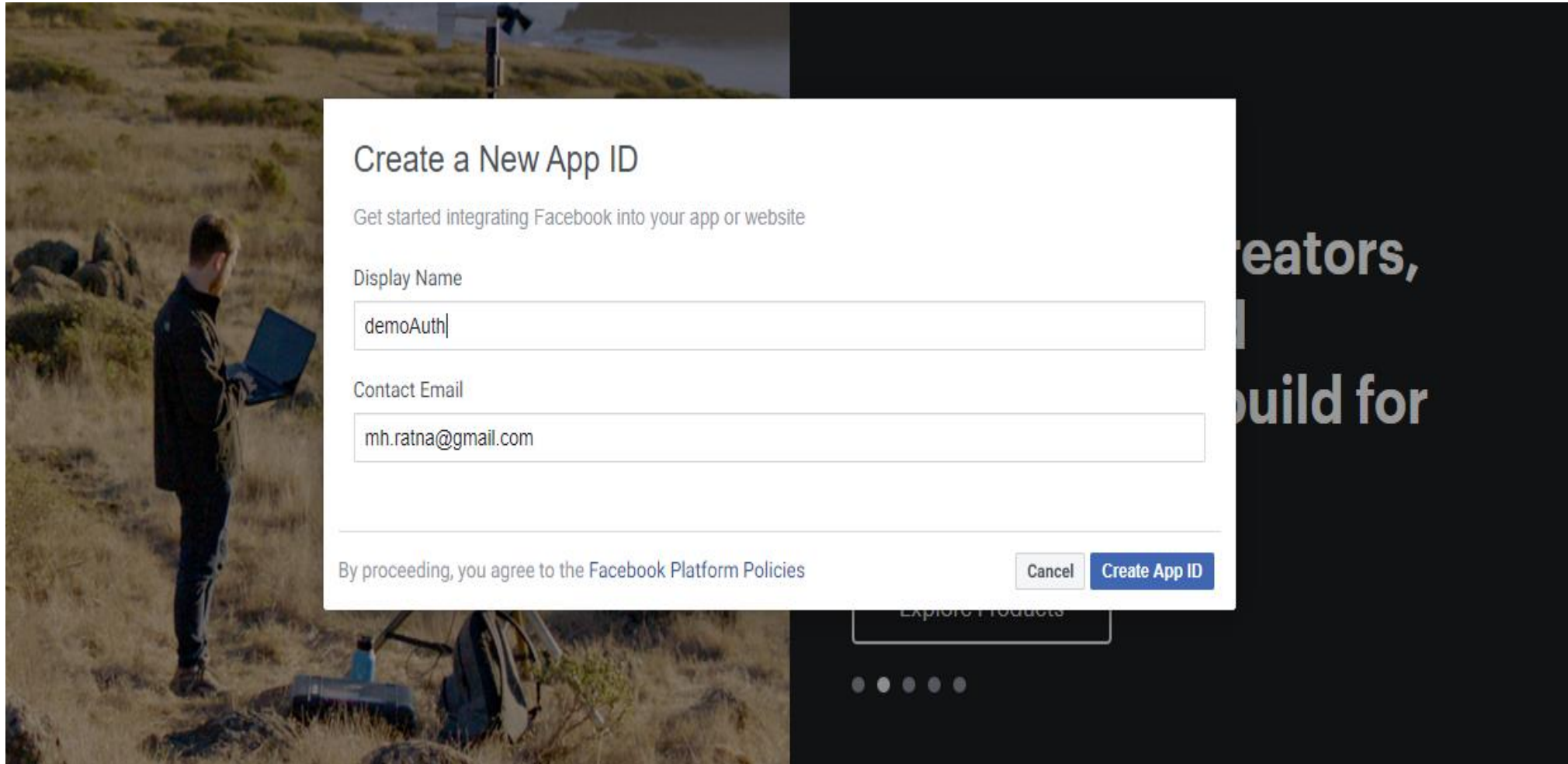
Step 1: Open the Facebook Developer Page and login with your facebook account

Step 2: Once the login is successful, go to myApps as shown in the screenshot and click on create App



Facebook Authentication

Step 3: Give a name to your app as shown in the screenshot and create App Id:



Create a New App ID

Get started integrating Facebook into your app or website

Display Name

demoAuth

Contact Email

mh.ratna@gmail.com

By proceeding, you agree to the Facebook Platform Policies

Cancel Create App ID

Facebook Authentication

Step 4: Copy App Id and Secret key which you can find in the basic setting of app.

facebook for developers

DocsToolsSupportMy Apps

Search developer documentation

demoAuth

APP ID: 448181669168974

OFF

Status: In Development

View Analytics

Help

Dashboard

Settings

Basic

Advanced

Roles

Alerts

App Review

PRODUCTS

App ID

448181669168974

App Secret

.....

Show

Display Name

demoAuth

Namespace

App Domains

Contact Email

mh.ratna@gmail.com

Privacy Policy URL

Privacy policy for Login dialog and App Details

Terms of Service URL

Terms of Service for Login dialog and App Details

App Icon (1024 x 1024)

1024 x 1024

Category

Choose a Category

Find out more information about app categories here

Activate Windows

Go to Settings to activate Windows.

Discard

Save Changes

Facebook Authentication

Step 5: Copy App Id and Secret key which you can find in the basic setting of app.

facebook for developers Docs Tools Support My Apps Search developer documentation

demoAuth APP ID: 448181669168974 OFF Status: In Development View Analytics Help

Dashboard Settings Basic Advanced Roles Alerts App Review PRODUCTS +

App ID: 448181669168974 App Secret: [Masked] Show

Display Name: demoAuth Namespace: [Empty]

App Domains: [Empty] Contact Email: mh.ratna@gmail.com

Privacy Policy URL: Privacy policy for Login dialog and App Details Terms of Service URL: Terms of Service for Login dialog and App Details

App Icon (1024 x 1024): [Placeholder Image] Category: Choose a Category Find out more information about app categories here

Business Use: This app uses Facebook tools or data to [Empty]

Activate Windows Go to Settings to activate Windows. Discard Save Changes

Facebook Authentication

Step 6: Paste app id and secret code in your firebase authentication option

demoAuth

APP ID: 448181669168974

OFF

Status: In Development

View Analytics

Help

Dashboard

Settings

Roles

Alerts

App Review

PRODUCTS

Facebook Login

Settings

Quickstart

Activity Log

Client OAuth Settings

Yes

Client OAuth Login

Enables the standard OAuth client token flow. Secure your application and prevent abuse by locking down which token redirect URIs are allowed with the options below. Disable globally if not used. [?]

Yes

Web OAuth Login

Enables web-based Client OAuth Login. [?]

No

Force Web OAuth Reauthentication

When on, prompts people to enter their Facebook password in order to log in on the web. [?]

Yes

Use Strict Mode for Redirect URIs

Only allow redirects that use the Facebook SDK or that exactly match the Valid OAuth Redirect URIs. Strongly recommended. [?]

Yes

Enforce HTTPS

Enforce the use of HTTPS for Redirect URIs and the JavaScript SDK. Strongly recommended. [?]

No

Embedded Browser OAuth Login

Enable webview Redirect URIs for Client OAuth Login. [?]

Valid OAuth Redirect URIs

Valid OAuth redirect URIs.

Facebook Authentication

Step 7: Copy URI of firebase project and paste in the facebook developer's OAuth setting in product

demoAuth

APP ID: 448181669168974

OFF

Status: In Development

View Analytics

Help

Dashboard

Settings

Roles

Alerts

App Review

PRODUCTS

Facebook Login

Settings

Quickstart

Activity Log

Client OAuth Settings

Yes

Client OAuth Login

Enables the standard OAuth client token flow. Secure your application and prevent abuse by locking down which token redirect URIs are allowed with the options below. Disable globally if not used. [?]

Yes

Web OAuth Login

Enables web-based Client OAuth Login. [?]

No

Force Web OAuth Reauthentication

When on, prompts people to enter their Facebook password in order to log in on the web. [?]

Yes

Use Strict Mode for Redirect URIs

Only allow redirects that use the Facebook SDK or that exactly match the Valid OAuth Redirect URIs. Strongly recommended. [?]

Yes

Enforce HTTPS

Enforce the use of HTTPS for Redirect URIs and the JavaScript SDK. Strongly recommended. [?]

No

Embedded Browser OAuth Login

Enable webview Redirect URIs for Client OAuth Login. [?]

Valid OAuth Redirect URIs

Valid OAuth redirect URIs.

Facebook Authentication

Step 8: All the settings are completely done now. Next implement the login of Facebook by using the firebase API, similar to Google.

```
.....  
login() {  
    return this.afAuth.auth.signInWithRedirect(new firebase.auth.FacebookAuthProvider());  
}  
}
```

*Thank
you*

WebStack Academy

#83, Farah Towers,
1st Floor, MG Road,
Bangalore – 560001

M: +91-809 555 7332

E: training@webstackacademy.com

WSA in Social Media:

