



Social Media APIs

Sayed Fathy



[illegible]

Social media **VS** Social network



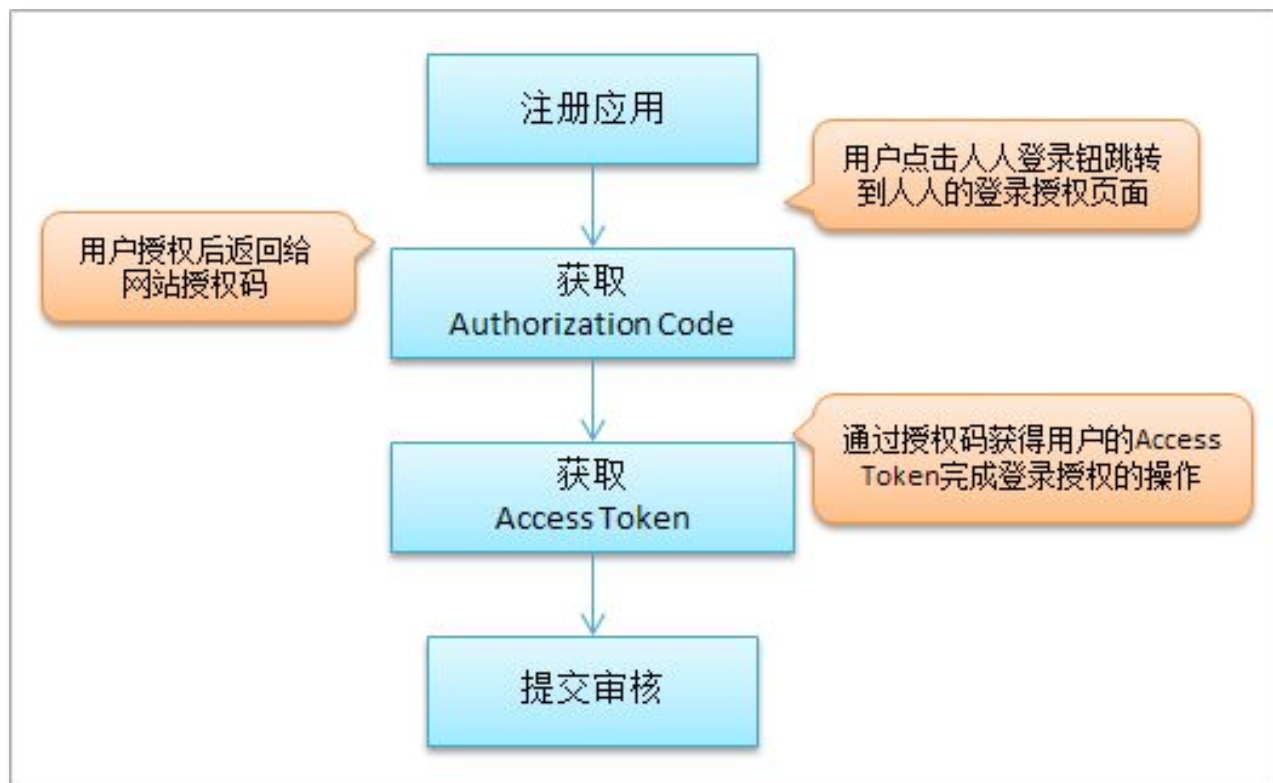
[illegible]

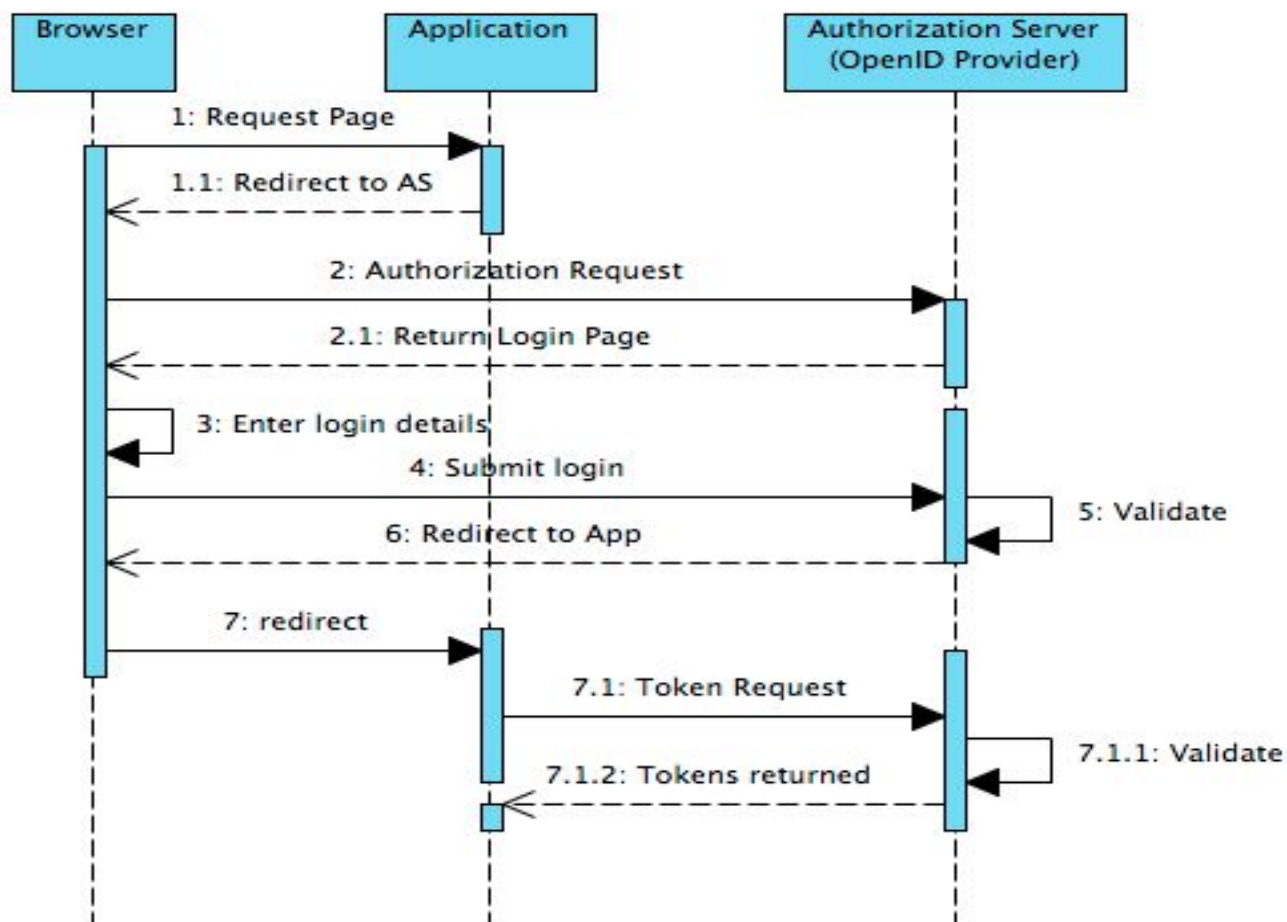
A dark blue circle is centered on a white background. Inside the circle, the text "{ api }" is written in a white, bold, sans-serif font. The text is centered horizontally and vertically within the circle. A thin, solid gold horizontal bar is located at the very bottom of the image, spanning the entire width.

{ api }

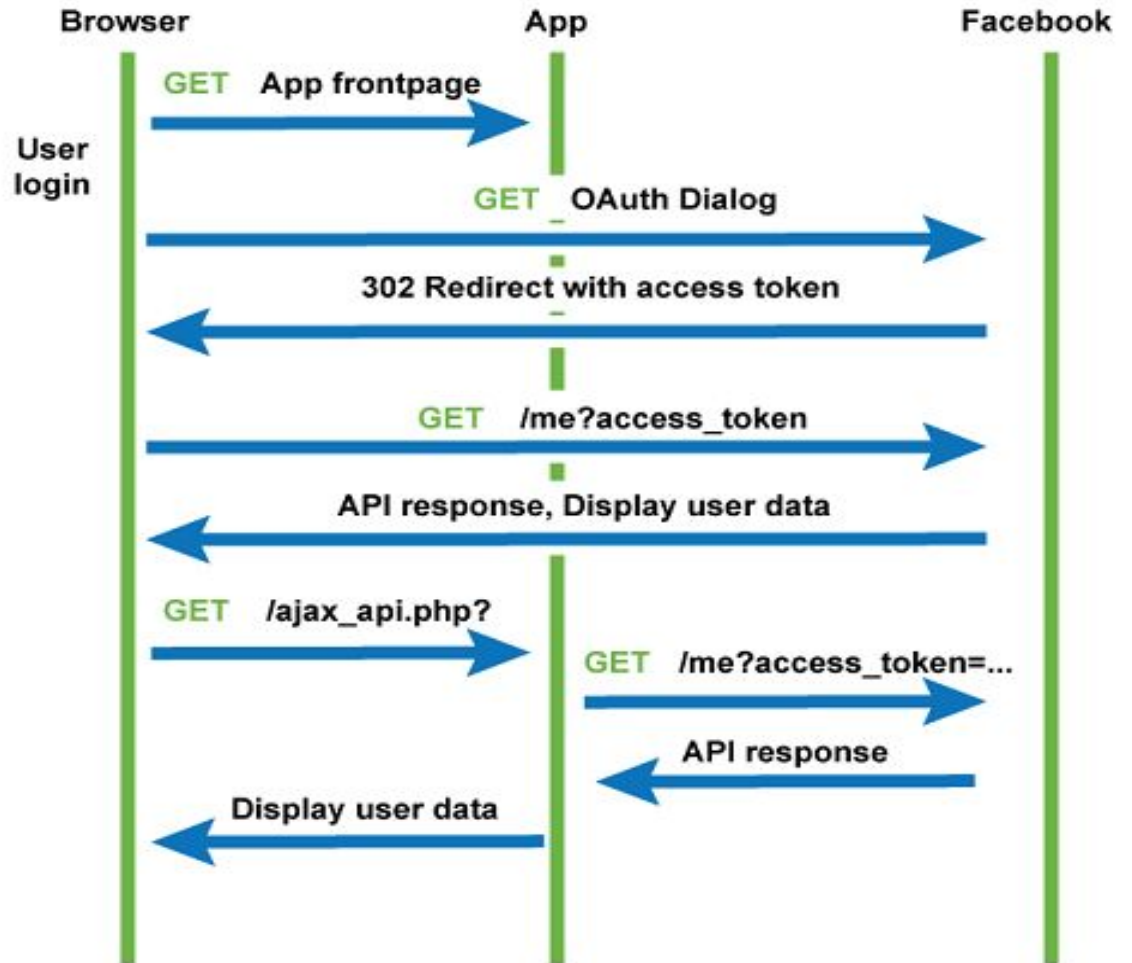


OAuth





Facebook OAuth scenario



Today

We are gonna explore
Facebook Graph API.

Facebook API

Create a facebook app to start using Facebook API.

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

1

Facebook API

Great!! now you are ready to use Facebook API.

2

this!

2

Step 2.1 downloading sdk

Download the suitable SDK you wanna use according to your preferred programming language.

Ex:

PHP, javascript, python, android, ios,...SDKs.

PHP SDK

The Facebook SDK for PHP provides a rich set of server-side functionality for accessing Facebook's server-side API calls.

Whether you're developing a website with Facebook login, creating a Facebook Canvas app or Page tab, the Facebook SDK for PHP does all the heavy lifting for you making it as easy as possible to deeply integrate into the Facebook platform.



Step 2.2 including sdk

```
require_once 'path_to_facebook-php-  
sdk/src/Facebook/autoload.php';
```

Step 2.3 loading app config

Before we can send requests to the Graph API, we need to load our app configuration into the `Facebook\Facebook` service.

```
$fb = new Facebook\Facebook([  
    'app_id' => '{app-id}',  
    'app_secret' => '{app-secret}',  
    'default_graph_version' => 'v2.5',  
]);
```

Note: You'll need to replace the `{app-id}` and `{app-secret}` with your Facebook app's ID and secret

Request Form

```
GET graph.facebook.com
```

```
/{node_id}?
```

```
fields={required_fields}
```

Request Form

```
GET graph.facebook.com
```

```
/me?
```

```
fields=albums.limit(5){name, photos.limit(2){name, picture,  
tags.limit(2)}},posts.limit(5)
```

Request Form

```
GET graph.facebook.com
```

```
/me?
```

```
fields=albums.limit(5),posts.limit(5)
```

Note:- To get the full list of Graph API endpoints, check this [link](#)

Facebook API Explorer

You can test all API through this [link](#)

PHP SDK

- Core API
- Authentication API
- Request and Response API
- Graph Nodes and Edges.

Facebook\Facebook

The Facebook\Facebook service class provides an easy interface for working with all the components of the SDK.

get(), post(), delete(), getDefaultAccessToken(), setDefaultAccessToken(), getRedirectLoginHelper()

```
$fb->get ( '/me' ) ;
```

```
#$fb is an object from Facebook\Facebook class
```


Facebook\Helpers\FacebookRedirectLoginHelper

The most commonly used helper is the `FacebookRedirectLoginHelper` which allows you to obtain a user access token from a redirect using a "Log in with Facebook" link.

You can obtain an instance of the `FacebookRedirectLoginHelper` from the `getRedirectLoginHelper()` method on the `Facebook\Facebook` service.

`getLoginUrl()`, `getLogoutUrl()`, `getAccessToken()`

```
$fb = new Facebook\Facebook([/* . . . */]);
```

```
$helper = $fb->getRedirectLoginHelper();
```

Facebook\Helpers\FacebookRedirectLoginHelper

```
$fb = new Facebook([ /* . . . */]);
```

```
$helper = $fb->getRedirectLoginHelper();
```

```
$permissions = ['email', 'user_likes']; // optional
```

```
$loginUrl = $helper->getLoginUrl('http://{your-website}/login-callback.php',  
$permissions);
```

```
echo '<a href="' . $loginUrl . '">Log in with Facebook!</a>';
```

Facebook\Authentication\AccessToken

Requests to the Graph API need to have an access token sent with them to identify the app, user and/or page that is making the request. The `Facebook\Authentication\AccessToken` entity represents an access token.

`getValue()`, `getExpiresAt()`, `isExpired()`

You can obtain an instance of the `AccessToken` from `getAccessToken()` method on the `Facebook\Helpers\FacebookRedirectLoginHelper` service.

Facebook\Authentication\AccessToken

```
$helper = $fb->getRedirectLoginHelper();
```

```
try {
```

```
    $accessToken = $helper->getAccessToken();
```

```
} catch(Facebook\Exceptions\FacebookResponseException $e) {
```

```
    // When Graph returns an error
```

```
    echo 'Graph returned an error: ' . $e->getMessage();
```

```
    exit;
```

```
}
```

Facebook\FacebookRequest

You can instantiate a new **FacebookRequest** entity directly by sending the arguments to the constructor.

```
$request = new FacebookRequest(  
    Facebook\FacebookApp $app,  
    string $accessToken,  
    string $method,  
    string $endpoint,  
    array $params  
);
```

Note:- To get the full list of Graph API endpoints, check this [link](#)

Facebook\FacebookResponse

After sending a request to the Graph API, the response will be returned in the form of a `Facebook\FacebookResponse` entity.

```
try {  
    $response = $fb->get('/me');  
} catch (Facebook\Exceptions\FacebookResponseException $e) {  
    // When Graph returns an error  
    echo 'Graph returned an error: ' . $e->getMessage();  
    exit;  
}  
  
var_dump($response);
```

instance methods

```
public Facebook\FacebookRequest getRequest();
```

```
public string getAccessToken();
```

```
public int getHttpStatusCode();
```

```
public array getHeaders();
```

```
public string getBody();
```

instance methods

```
public Facebook\GraphNodes\GraphNode getGraphNode();
```

```
public Facebook\GraphNodes\GraphUser getGraphUser();
```

```
public Facebook\GraphNodes\GraphEdge getGraphEdge();
```


Facebook\GraphNode\GraphNode

A `Facebook\GraphNode\GraphNode` is a collection that represents a node returned by the Graph API.

`GraphNode`'s are obtained from a `Facebook\FacebookResponse` object which represents an HTTP response from the Graph API.

`getField()`, `getFieldNames()`, `asArray()`, `asJson()`, `getLink()`

Facebook\GraphNode\GrpahUser

The `GraphUser` collection represents a `User` Graph node.

```
// Returns a `Facebook\FacebookResponse` object
```

```
$response = $fb->get('/something');
```

```
$graphNode = $response->getGraphNode();
```

```
$user = $response->getGraphUser();
```

```
// User example
```

```
echo $graphNode->getField('name'); // From GraphNode
```

```
echo $user->getName(); // From GraphUser
```

```
// Location example
```

```
echo $graphNode->getField('country'); // From GraphNode
```