GUIDELINE

INTEGRATE LAOID LOGIN

# OVERVIEW

Phương thức Đăng nhập bằng LaoID, tương tự như đăng nhập bằng Google hay facebook, là cách thức nhanh chóng và thuận tiện để người dùng truy cập và sử dụng các dịch vụ trong hệ sinh thái các dịch vụ do Unitel cung cấp hoặc các dịch vụ có hợp tác với Unitel.

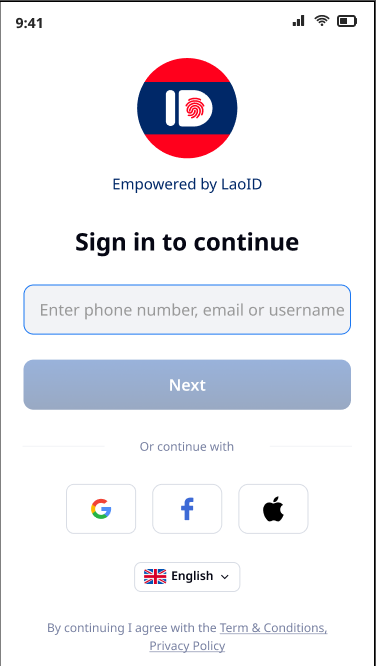
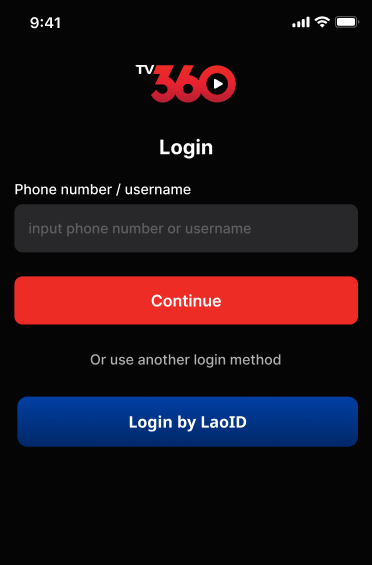
Phương thức này đáp ứng việc đăng nhập trên đủ các nền tảng, bao gồm iOS, Android, web, ứng dụng trên máy tính và các thiết bị như TV thông minh, thiết bị kết nối Internet.

Phương thức Đăng nhập bằng LaoID được sử dụng trong 2 trường hợp, đó là xác thực và yêu cầu quyền truy cập vào dữ liệu của người dùng. Các hệ thống có thể dùng phương thức Đăng nhập bằng LaoID **để xác thực** hoặc **để xác thực và truy cập dữ liệu người dùng**.

Sau khi đăng nhập thành công và được xác nhận là có quyền để truy cập dữ liệu người dùng, hệ thống sử dụng LaoID Login có thể sử dụng mã bí mật để lấy được các thông tin như sau:

* Các thông tin cơ bản về người dùng bao gồm: tên, ngày tháng năm sinh, giới tính, địa chỉ
* Avatar của người dùng hiện có trên LaoID
* Thông tin liên hệ bao gồm các số điện thoại và email có liên kết
* Thông tin về các phương thức thanh toán khả dụng

Các dịch vụ hoặc ứng dụng tích hợp login bằng LaoID có thể thiết kế nút bấm riêng hoặc lập trình để khi mở ứng dụng thì vào thẳng màn hình login bằng LaoID như hình ảnh minh họa dưới đây:



Open wapview or redirect to sso.laoid.net

Open app then open wapview or redirect to sso.laoid.net

# MÔ TẢ GIẢI PHÁP LOGIN BY LAOID

Flowchart mô tả tương tác giữa người dùng các hệ thống như dưới đây:

A diagram of a service provider

Description automatically generated

## Diễn giải các bước xác thực qua LaoID

Khi người dùng chọn đăng nhập trên trang dịch vụ, phía dịch vụ sẽ redirect người dùng tới LaoID để đăng nhập. Sau khi người dùng thực hiện đăng nhập thành công trên LaoID thì LaoID sẽ redirect người dùng trở lại ứng dụng hoặc trang web của dịch vụ kèm theo thông tin người dùng.

## Các thông tin cần chuẩn bị trước khi thực hiện tích hợp (before to start)

Để tích hợp đăng nhập bằng LaoID, đơn vị phát triển web/app cần phải có 2 thông tin quan trọng được cung cấp bởi LaoID, bao gồm:

* ClientID
* ClientSecret

Ngoài ra, phải khai báo về địa chỉ URL của dịch vụ trên hệ thống LaoID để callback lại sau khi login thành công.

Để có thể khai báo các thông tin về URL tích hợp login và URL thực hiện callback, đơn vị phát triển có thể thực hiện theo 2 cách:

* Đối với đơn vị không có account trên hệ thống LaoCRM thì cần liên hệ trực tiếp với đầu mối hỗ trợ của LaoID thông qua email [support@laoid.net](mailto:support@laoid.net) để gửi các thông tin về URL và đồng thời để nhận được thông tin liên quan đến ClientID và ClientSecrect tương ứng với các URL kể trên, phục vụ cho việc tích hợp.
* Đối với các đơn vị đã có account trên hệ thống LaoCRM thì tại màn hình khai báo một service mới, đơn vị phát triển cần enable chức năng login với LaoID để có thể khai báo các URL liên quan và sau đó nhận được thông tin về ClientID và ClientSecret trực tiếp trên giao diện CMS của LaoCRM này. Mô tả như hình ảnh dưới đây:

A screenshot of a login box

Description automatically generated

# TÍCH HỢP SSO LAOID

Để tích hợp LaoID, đơn vị phát triển có thể thực hiện theo phương thức tự động khởi tạo hoặc gọi hàm init của SDK.

## Cách 1: Tự động khởi tạo

SDK sẽ tự động khởi tạo dựa vào thông tin định nghĩa trong thẻ meta. Tại thẻ head trong code html bổ sung thêm, giá trị “content” được để là tên của dịch vụ hoặc ứng dụng sẽ cung cấp cho người dùng; giá trị trường “laoid-signin-use-callback-uri” được set là true để điều hướng người dùng về trang web của đơn vị phát triển thì (callback uri)

<meta

name="laoid-signin-client-id"

content="[Service\_Name]"

/>

<meta

name="laoid-signin-redirect-uri"

content="[Service\_Name]"

/>

<meta

name="laoid-signin-use-callback-uri"

content="true"

/>

<script

type="text/javascript"

src="https://sso.laoid.net/laoid.auth.js"

async

/>

## Cách 2: Khởi tạo SDK bằng cách gọi hàm init của SDK.

Tại thẻ head trong html thì thêm script:

<script

type="text/javascript"

src="https://sso.laoid.net/laoid.auth.js"

async

/>

Sau đó thêm một đoạn code sau trong file javascript để gọi hàm khởi tạo

window.LaoIdSSO.init(

client\_id,

redirect\_uri,

use\_callback\_uri

)

Trong HTML thêm nút đăng nhập bằng LaoID

<button id="laoid-signin">Login by LaoId</button>

# THỦ TỤC MỞ TRANG WAP LOGIN BY LAOID

Khi người dùng click vào nút đăng nhập bằng LaoID, SDK sẽ tự thực hiện việc mở đến trang login của LaoID với thông tin như sau, lưu ý đơn vị phát triển không cần làm việc mở link này thủ công.

https://sso.laoid.net/login?client\_id=${laoid-signin-client-id}&redirect\_uri=${laoid-signin-callback-uri}&use\_callback\_uri={laoid-signin-use-callback-uri}

Trong đó:

|  |  |  |
| --- | --- | --- |
| **Tên** | **Mô tả** | **Kiểu dữ liệu** |
| laoid-signin-client-id | là thông tin **client\_id** đã được cấp trước đó | string |
| laoid-signin-callback-uri | Là thông tin **callback\_uri** đã được cấp trước đó | string |
| laoid-signin-use-callback-uri | Là trường để xác định có điều hướng người dùng về trang web của đơn vị phát triển hay không? | boolean |

Nếu các thông tin là chính xác thì wap login sso.laoid.net được mở ra và người dùng thực hiện các bước login hoặc tạo tài khoản trực tiếp với hệ thống LaoID

Nhận thông tin kết quả xác thực login của người dùng từ LaoID

Tuỳ thuộc vào việc đơn vị phát triển đặt giá trị trường laoid-signin-use-callback-uri thì sẽ sử dụng 2 cách khác nhau để lấy kết quả

## Cách 1: Giá trị bằng true

Khi login thành công trên LaoID thì LaoID sẽ điều hướng người dùng đến địa chỉ URL callback đã khai báo trước đó và kèm theo thông tin như sau:

${laoid-signin-callback-uri}?authorization\_code=${authorization\_code}

Đơn vị phát triển sẽ thực hiện nhận dạng đường link trả về này và tách authorization\_code.

Trường hợp nếu sử dụng LaoID chỉ cho mục đích xác thực người dùng và không cần lấy thông tin người dùng thì việc tích hợp đến bước này coi như là thành công.

## Cách 2: Giá trị bằng false

Khi login thành công trên LaoID thì LaoID sẽ đóng tab lại và bắn authorization\_code qua code javascript, đơn vị phát triển sẽ tạo hàm lắng nghe kết quả trả về như sau:  
  
const onMessage = (event) => {

if (event.origin !== "https://sso.laoid.net") return;

const {message, data} = event.data

TODO handleData

};

window.addEventListener("message", onMessage, false);

Trong **event.data** sẽ chứa 2 thông tin:

* message: chứa thông báo
  + login\_success: Đăng nhập thành công
  + login\_fail: Xác thực *laoid-signin-client-id, laoid-signin-redirect-uri* thất bại
* data: chứa **authorizationCode** bên SSO trả về

# TRUY VẤN THÔNG TIN NGƯỜI DÙNG SAU ĐĂNG NHẬP

Sau khi login xác thực thành công, nếu đơn vị phát triển muốn truy vấn thông tin người dùng đã đăng nhập ở bước trước đó thì có thể sử dụng authorization\_code để lấy accessToken và sau đó là sử dụng AccessToken để lấy thông tin người dùng. Lưu đồ mô tả các bước thực hiện như sau:

**Service Provider**

**SSO Server (LaoID)**

**Request** to **get AccessCode**

**Request** to **Check AccessCode**

Request to **get user\_info**

Response with AccessCode

Response with result of checking

Response with infomation of enduser

**Not require**

Just for checking

Việc giao tiếp giữa ứng dụng của đơn vị phát triển và hệ thống LaoID để truy vấn dữ liệu được thực hiện thông qua các API giữa backend của 2 hệ thống để đảm bảo bảo mật và tính chính xác của dữ liệu. Giao tiếp này được mô tả cụ thể qua 3 bước như sau.

## **Step 1**: Giao tiếp giữa 2 backend để lấy thông tin về AccessToken

Hệ thống của đơn vị phát triển sẽ gọi API sang LaoID để request lấy thông tin về Token, cụ thể:

**URL**: [https://sso.laoid.net/api/v1/third-party/verify](https://laoid.net/api/v1/third-party/verify)

* Method: POST
* Content Type: application/json

**Input**:

|  |
| --- |
| **Header**   * X-Accept-Language: Ngôn ngữ của người dùng trong hệ thống   **Param**   * N / A   **Request Body**   * **code**: Authorization code nhận được sau khi đăng nhập * **clientId**: được được cấp trước đó * **clientSecret**: đã được cấp trước đó |

**Output**

|  |
| --- |
| {  "success": true,  "message": "Get access token successfully",  "data": {  "accessToken": "eyJhbGciOiJIUzUxMiJ9..6VirfUQs26hAgNUFk8t7M5d83N\_lbzUbhJ9UhIbRLAnUMLKFUZa5NQrlVSosJACp4rq7rUKZzJfszOkxwQtkfg",  "expiresIn": 1209600,  "idToken": "eyJhbGciOiJIUzUxMiJ9. ",  "tokenType": "bearer"  },  "statusCode": "THIRDPARTY1206"  } |
| **Response Code:**   * THIRDPARTY1206: Lấy access token thành công * THIRDPARTY0207: Lấy access token thất bại * THIRDPARTY0208: Thông tin client không hợp lệ (clientId hoặc clientSecret không hợp lệ) * THIRDPARTY0209: Authorization code không hợp lệ * THIRDPARTY0217: Authorization code không phải của client đang xác thực |

## **Step 2**: Giao tiếp giữa 2 backend để kiểm tra AccessToken (Tùy chọn - không bắt buộc)

Hệ thống của đơn vị phát triển sẽ gọi API sang LaoID để check thông tin về accessToken, cụ thể:

**URL**: [https://sso.laoid.net/api/v1/third-party/token-verifier](https://laoid.net/api/v1/third-party/token-verifier)

* Method: POST
* Content Type: application/json

**Input**:

|  |
| --- |
| **Header**   * X-Accept-Language: Ngôn ngữ của người dùng trong hệ thống   **Param**   * N / A   **Request Body**   * **clientId**: được được cấp trước đó * **clientSecret**: đã được cấp trước đó * **AccessToken**: Thông tin về token đã nhận từ response trước đó |

**Output**

|  |
| --- |
| {  "success": true,  "message": "Access token is valid",  "data": {  " isVerified ": True  },  "statusCode": "THIRDPARTY1208"  } |
| **Response Code:**   * THIRDPARTY1208: Access token hợp lệ * THIRDPARTY1209: Access token không hợp lệ * THIRDPARTY0222: Thông tin client không hợp lệ (clientId hoặc clientSecret không hợp lệ) * THIRDPARTY0223: Xác thực Access token thất bại |

## **Step 3**: Giao tiếp giữa 2 backend để lấy thông tin người dùng

Hệ thống của đơn vị phát triển sẽ gọi API sang LaoID để lấy thông tin người dùng, cụ thể:

**URL**: [https://sso.laoid.net/api/v1/third-party/me](https://laoid.net/api/v1/third-party/me)

* Method: GET
* Content Type: application/json

**Input:**

|  |
| --- |
| **Header**   * X-Accept-Language: Ngôn ngữ của người dùng trong hệ thống * Authorization:Access token của người dùng trong hệ thống (Dạng: “Bearer ” + Access token) * x-api-key:Chính là Client ID đã được cấp trước đó   **Param**   * N / A |

**Output**

|  |
| --- |
| {  "success": true,  "message": "Get user information successfully",  "data": {  "id": "10000000001",  "username": "Quanghuy270200",  "firstName": "Quang",  "lastName": "Anh",  "avatar":"https://demo-sso-image.tinasoft.io/sso/avatar/10000000001/1717380883031.jpg",  "banner": null,  "dateOfBirth": "2002-12-31",  "gender": null,  "publicId": null,  "language": "vi",  "email": [  {  "email": "daoquanghuy2702@gmail.com",  "primary": true  }  ],  "phoneNumber": [  {  "phoneNumber": "84987654327",  "primary": true  }  ],  "province": null,  "country": 35,  "provinceName": null,  "countryName": "Bulgaria",  "hasPassword": true,  "paymentMethods": [  {  "id": "cbe95b2c-70cd-4ff6-86f8-599743029281",  "account": "8560010000001",  "paymentOrder": 1,  "type": "MOBILE\_BALANCE",  "loyaltyAccount": false  },  {  "id": "439147a3-56c6-4ac3-bdae-af4b890a5481",  "account": "8560010000001",  "paymentOrder": 2,  "type": "U\_POINT",  "loyaltyAccount": true  }  ]  },  "statusCode": "USER1200"  } |
| **Response Code:**   * USER1200: Lấy thông tin người dùng thành công * USER0200: Lấy thông tin người dùng thất bại * USER0224: Xảy ra lỗi trong quá trình lấy thông tin người dùng |