**API Specification Doc**

**(*Zalo Fake App*)**

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author | Description |
| 1.0 | Feb-25-2024 | Bui Tri Thuc | Authenticate and Register |
| 1.0.1 | March-05-2024 | Bui Tri Thuc | Change the params (device\_id) of Login API request |
| 1.1 | March-26-2024 | Bui Tri Thuc | Add feature : update user profile |

# Authentication Features

## 1. Login

* Authenticate the user with the system and obtain the token

### Request

|  |  |
| --- | --- |
| **Method** | **URL** |
| **POST** | /api/auth/login |

|  |  |  |
| --- | --- | --- |
| **Type** | **Params** | **Values** |
| POST  POST  POST | phone  password  device\_id [optional] | string  string  string ( get device id ) |

### Response

|  |  |  |
| --- | --- | --- |
| **Status** | **Response** | **Note** |
| 200 | {  "user": {  "profile": {  "name": "Tester02"  },  "phone": "0909878765",  "createdAt": "2024-02-28T08:18:54.887Z",  "friendList": [],  "groupList": [],  "posts": [],  "moments": [],  "\_\_v": 0  },  "accessToken": < Access Token >  } | Login on Web**(2)** |
| 200 | {  "user": {  "profile": {  "name": "Tester02"  },  "phone": "0909878765",  "createdAt": "2024-02-28T08:18:54.887Z",  "friendList": [],  "groupList": [],  "posts": [],  "moments": [],  "\_\_v": 0  },  "accessToken": < Access Token >,  "refreshToken": <Refresh Token>  } | Login on Mobile |
| 401 | {"message":"Invalid phone or password"} |  |
| 400 | {"message":"error"} |  |
| 500 | {"message":"Something went wrong. Please try again later."} |  |

***(1) :*** If login form web applications, device\_id is not required, server will be genarate and manager this base on user IP Address

***(2) :*** If login form web applications, refresh token will be add to cookies without return it in response body.

## 2. Logout

* Logout the user with the system and remove the token

### Request

|  |  |
| --- | --- |
| **Method** | **URL** |
| **POST** | /api/auth/logout |

|  |  |  |
| --- | --- | --- |
| **Type** | **Params** | **Values** |
| HEADER.Cookie | refreshToken | string |

### Response

|  |  |  |
| --- | --- | --- |
| **Status** | **Response** | **Note** |
| 200 | {"message":"Logout successfully"} |  |
| 403 | {"message":"You’re not authenticate !"} | Missing refresh token value |
| 403 | {"message":"Refresh token is not valid !"} |  |
| 500 | {"message":"Something went wrong. Please try again later."} |  |

## 3. Register

* Register the user and account of this user with the system

### Request

|  |  |
| --- | --- |
| **Method** | **URL** |
| **POST** | /api/auth/register |

|  |  |  |
| --- | --- | --- |
| **Type** | **Params** | **Values** |
| POST  POST  POST | phone  password  name | string  string  string |

### Response

|  |  |  |
| --- | --- | --- |
| **Status** | **Response** | **Note** |
| 201 | {  "user": {  "phone": <phone of user sign up>,  "profile": {  "name": <Name of user sign up>  }  }  } |  |
| 400 | {"message":<error while create user>} |  |
| 409 | {"message":"Phone number already exists"} |  |
| 500 | {"message":"Something went wrong. Please try again later."} |  |

## 4. Refresh Token

* To ensure security, an access token only remains valid for 10 minutes, after which it needs to be refreshed whenever it expires

### Request

|  |  |
| --- | --- |
| **Method** | **URL** |
| **POST** | /api/auth/refreshToken |

|  |  |  |
| --- | --- | --- |
| **Type** | **Params** | **Values** |
| HEADER.Cookie | refreshToken | string |

### Response

|  |  |  |
| --- | --- | --- |
| **Status** | **Response** | **Note** |
| 200 | {  "newAccessToken": <New access token>  } | Request form Web**(2)** |
| 200 | {  "newAccessToken": <New access token>,  "newRefreshToken": < New Refresh token>  } | Request form Mobile |
| 403 | {"message":"You’re not authenticated !"} | Missing refresh token value |
| 403 | {"message":"Refresh Token í not valid !"} |  |
| 500 | {"message":"Something went wrong. Please try again later."} |  |

***(2) :*** If request form web applications, the new refresh token will be add to cookies without return it in response body.

# User Features

## Update profile

### Request

|  |  |
| --- | --- |
| **Method** | **URL** |
| **POST** | /api/users/update-profile |

|  |  |  |
| --- | --- | --- |
| **Type** | **Params** | **Values** |
| POST  POST  POST  POST  POST | userId  name  email  gender  dob | string  string  string  string (enum {male, female, orther})  string (format {yyyy-MM-dd}) |

### Response

|  |  |  |
| --- | --- | --- |
| **Status** | **Response** | **Note** |
| 200 | {  "message": "User updated successfully",  "user": {  "profile": {  "avatar": {  "url": "<image url>",  "public\_id": "iamge public id"  },  "background": {  "url": "<image url>",  "public\_id": "iamge public id"  },  "name": "<user’s name>",  "dob": "<user’s dob>",  "gender": "male"  },  "phone": "<user’s phone>",  "createdAt": "<Date>",  "\_\_v": 1,  "lastActive": "<Date>",  "email": "user’s email"  }  } |  |
| 404 | {"message":"User not found"} |  |
| 400 | {"message":"Can't update this user"} |  |
| 500 | {"message":"Something went wrong. Please try again later."} |  |

## Update avatar

### Request

|  |  |
| --- | --- |
| **Method** | **URL** |
| **POST** | /api/users/update-avatar |

|  |  |  |
| --- | --- | --- |
| **Type** | **Params** | **Values** |
| POST  POST | userId  avatar | string  file |

### Response

|  |  |  |
| --- | --- | --- |
| **Status** | **Response** | **Note** |
| 200 | {  "message": "Avatar uploaded successfully",  "avatar": {  "url": "<new image url>",  "public\_id": "new image url"  }  } |  |
| 404 | {"message":"User not found"} |  |
| 400 | {"message":"userId is required"} | UserId not found in request body |
| 400 | {"message":"Image is required"} | Image file not found in request body |
| 400 | {"message":"Can't update avatar"} |  |
| 500 | {"message":"Something went wrong. Please try again later."} |  |

## Update background

### Request

|  |  |
| --- | --- |
| **Method** | **URL** |
| **POST** | /api/users/update-background |

|  |  |  |
| --- | --- | --- |
| **Type** | **Params** | **Values** |
| POST  POST | userId  background | string  file |

### Response

|  |  |  |
| --- | --- | --- |
| **Status** | **Response** | **Note** |
| 200 | {  "message": "Background uploaded successfully",  "avatar": {  "url": "<new image url>",  "public\_id": "new image url"  }  } |  |
| 404 | {"message":"User not found"} |  |
| 400 | {"message":"userId is required"} | UserId not found in request body |
| 400 | {"message":"Image is required"} | Image file not found in request body |
| 400 | {"message":"Can't update background"} |  |
| 500 | {"message":"Something went wrong. Please try again later."} |  |

## Get user information

### Request

|  |  |
| --- | --- |
| **Method** | **URL** |
| **GET** | /api/users/get/phone/{phone number} |
| **GET** | /api/users/get/uid/{user id} |

### Response

|  |  |  |
| --- | --- | --- |
| **Status** | **Response** | **Note** |
| 200 | {  "user": {  "profile": {  "avatar": {  "url": "<image url>",  "public\_id": "image public id"  },  "background": {  "url": "<image url>",  "public\_id": "image public id"  },  "name": "<user’s name>",  "dob": "<user’s dob>",  "gender": "male"  },  "email": "<user’s email>",  "phone": "<user’s phone>"  }  } |  |
| 404 | {"message":"User not found"} |  |
| 400 | {"message":"Can’t not get this user"} |  |
| 500 | {"message":"Something went wrong. Please try again later."} |  |

# Conventions

* **Client** - Client application.
* **Status** - HTTP status code of response.
* **Type –** HTTP Request Method (GET, POST, PUT, PATH, DELETE…)
* All response are in JSON format.
* All request parameters are mandatory unless explicitly marked as [optional]

## Status Codes

All status codes are standard HTTP status codes. The below ones are used in this API.

2XX - Success of some kind

4XX - Error occurred in client’s part

5XX - Error occurred in server’s part

|  |  |
| --- | --- |
| **Status Code** | **Description** |
| 200 | OK |
| 201 | Created |
| 202 | Accepted (Request accepted, and queued for execution) |
| 400 | Bad request |
| 401 | Authentication failure |
| 403 | Forbidden |
| 404 | Resource not found |
| 405 | Method Not Allowed |
| 409 | Conflict |
| 412 | Precondition Failed |
| 413 | Request Entity Too Large |
| 500 | Internal Server Error |
| 501 | Not Implemented |
| 503 | Service Unavailable |