

CUJAE LDAP API

Overview

API para la administración de usuarios LDAP de la CUJAE

Tags

Authentication

API para la autenticación de usuarios

Users

API para la administración de usuarios LDAP

Groups

API para la administración de grupos LDAP

Profile

API para la administración del perfil de usuario

Recovery Password

API para la recuperación de contraseña

Update Password

API para la actualización de contraseña

Reset Password

API para el reseteo de contraseña

Logs

API para la administración de logs

DN

API para la administración de DNs

Paths

***GET* /api/v1/searchByDN Search by Distinguished Name (DN).**

Retrieve information by providing a Distinguished Name (DN).

Parameters

Type	Name	Description	Schema
query	baseDN <i>required</i>	The Distinguished Name (DN) to search for.	string

Responses

Code	Description	Links
200	Successfully retrieved information. <i>Content</i> application/json	No Links
400	Bad Request. Base DN is missing.	No Links
401	Unauthorized. Requires admin role.	No Links
500	Internal Server Error. Failed to fetch information.	No Links

Type	Name	Scopes
apiKey	BearerAuth	

***GET* /api/v1/groups/byName/{group}** Get a LDAP group by its name.

Retrieve information about an LDAP group by specifying its name in the URL path. This endpoint requires authentication to access group information.

Parameters

Type	Name	Description	Schema
path	group <i>required</i>	The name of the LDAP group to retrieve.	string

Responses

Code	Description	Links
200	LDAP group information retrieved successfully.	No Links
401	Unauthorized. The user is not authenticated.	No Links
500	An error occurred while fetching group information.	No Links

Type	Name	Scopes
apiKey	BearerAuth	

***POST* /api/v1/groups/getChilds** Get child groups of a specified base DN.

Retrieve child groups of a specified base DN. This endpoint is restricted to administrators and requires authentication.

Responses

Code	Description	Links
200	Child groups retrieved successfully.	No Links
401	Unauthorized. The user is not authenticated.	No Links
403	Forbidden. The user does not have sufficient privileges to access this endpoint.	No Links
500	An error occurred while fetching child groups.	No Links

Type	Name	Scopes
apiKey	BearerAuth	

***POST* /api/v1/groups/byType/{type}** Get a group by type.

Retrieve a group by its type using the specified DN. This endpoint requires authentication.

Parameters

Type	Name	Description	Schema
path	type <i>required</i>	The type of the group to retrieve.	string

Responses

Code	Description	Links
200	Group retrieved successfully by type.	No Links
401	Unauthorized. The user is not authenticated.	No Links
500	An error occurred while fetching the group by type.	No Links

Type	Name	Scopes
apiKey	BearerAuth	

***GET* /api/v1/profile** Get the user profile.

Retrieve the user profile based on the UID (User ID) contained in the JSON Web Token (JWT) passed in the request headers. The endpoint requires authentication and user role permissions to access the profile information.

Responses

Code	Description	Links
200	User profile retrieved successfully.	No Links
401	Unauthorized. The user is not authenticated.	No Links
403	Forbidden. The user does not have the required permissions to access the profile.	No Links
500	An error occurred while retrieving the user profile.	No Links

Type	Name	Scopes
apiKey	BearerAuth	

PUT /api/v1/profile Update the user profile.

Update the user profile based on the provided information in the request body. This endpoint requires authentication and user role permissions to access and modify the profile information.

Responses

Code	Description	Links
200	User profile updated successfully.	No Links
400	Bad request. Invalid or missing parameters in the request body.	No Links
401	Unauthorized. The user is not authenticated.	No Links
403	Forbidden. The user does not have the required permissions to update the profile.	No Links
500	An error occurred while updating the user profile.	No Links

Type	Name	Scopes
apiKey	BearerAuth	

GET /api/v1/users Get a list of users.

Retrieve a list of users based on query parameters. You can filter the results by providing one or more of the following query parameters.

Parameters

Type	Name	Description	Schema
query	lastName <i>optional</i>	Filter users by last name.	string
query	country <i>optional</i>	Filter users by country.	string

Type	Name	Description	Schema
query	career <i>optional</i>	Filter users by career.	string
query	workerID <i>optional</i>	Filter users by worker ID.	string
query	userStatus <i>optional</i>	Filter users by status.	string
query	mail <i>optional</i>	Filter users by mail.	string
query	skinColor <i>optional</i>	Filter users by skin color.	string
query	displayName <i>optional</i>	Filter users by display name.	string
query	dn <i>optional</i>	Filter users by DN.	string
query	educationalCategory <i>optional</i>	Filter users by educational category.	string
query	maildrop <i>optional</i>	Filter users by maildrop.	string
query	studentYear <i>optional</i>	Filter users by student year.	string
query	UJC <i>optional</i>	Filter users by UJC.	string
query	uid <i>optional</i>	Filter users by UID.	string
query	workercontract <i>optional</i>	Filter users by worker contract.	string
query	orgRole <i>optional</i>	Filter users by organizational role.	string
query	userInformation <i>optional</i>	Filter users by information.	string
query	limit <i>optional</i>	The number of items per page.	integer
query	sn <i>optional</i>	Filter users by SN.	string

Type	Name	Description	Schema
query	email <i>optional</i>	Filter users by email.	string
query	area <i>optional</i>	Filter users by area.	string
query	schoolLevel <i>optional</i>	Filter users by school level.	string
query	sedeMunicipio <i>optional</i>	Filter users by municipality.	string
query	CI <i>optional</i>	Filter users by CI.	string
query	ou <i>optional</i>	Filter users by organizational unit.	string
query	sex <i>optional</i>	Filter users by sex.	string
query	studentClassGroup <i>optional</i>	Filter users by class group.	string
query	userCondition <i>optional</i>	Filter users by condition.	string
query	nameInstitution <i>optional</i>	Filter users by institution name.	string
query	userYears <i>optional</i>	Filter users by years of service.	string
query	cn <i>optional</i>	Filter users by CN.	string
query	workArea <i>optional</i>	Filter users by work area.	string
query	objectName <i>optional</i>	Filter users by object name.	string
query	page <i>optional</i>	The page number.	integer
query	scientificCategory <i>optional</i>	Filter users by scientific category.	string
query	username <i>optional</i>	Filter users by username.	string

Responses

Code	Description	Links
200	A list of users.	No Links
500	An error occurred.	No Links

GET /api/v1/group/{group} Get a list of users in a specific group.

Retrieve a list of users in the specified group based on query parameters.

Parameters

Type	Name	Description	Schema
query	limit <i>optional</i>	The number of items per page.	integer
query	page <i>optional</i>	The page number.	integer
path	group <i>required</i>	The group name for which to retrieve users.	string

Responses

Code	Description	Links
200	A list of users in the group.	No Links
500	An error occurred.	No Links

POST /api/v1/users/baseDN Get a list of users based on a custom baseDN.

Retrieve a list of users based on a custom baseDN. You can specify the baseDN in the request body to filter the results. Optionally, you can provide additional query parameters to further refine the search.

Parameters

Type	Name	Description	Schema
query	limit <i>optional</i>	The number of items per page.	integer
query	page <i>optional</i>	The page number.	integer

Responses

Code	Description	Links
200	A list of users based on the custom baseDN.	No Links
400	Bad request. The baseDN parameter is missing in the request body.	No Links
500	An error occurred.	No Links

PUT /api/v1/users/{username} Update a user's attributes.

Update a user's attributes by specifying the username in the URL path. You can provide the attribute name (**att**) and the new value (**value**) in the request body to perform the update.

Parameters

Type	Name	Description	Schema
path	username <i>required</i>	The username of the user to update.	string

Responses

Code	Description	Links
200	User attributes updated successfully.	No Links
400	Bad request. The username parameter is missing or invalid.	No Links
500	An error occurred while updating user attributes.	No Links

POST /api/v1/users/modify-ldap Modify LDAP user attributes.

Modify LDAP user attributes by specifying the DN (Distinguished Name) and the new attribute values. You can provide a DN in the request body and an object containing attributes to modify. The attributes are provided as key-value pairs, where the key is the attribute name and the value is the new value to set. All specified attributes will be replaced with the new values.

Responses

Code	Description	Links
200	LDAP user attributes modified successfully.	No Links
400	Bad request. The dn or attributes parameter is missing or invalid.	No Links
500	An error occurred while modifying LDAP user attributes.	No Links

POST /api/v1/users/newUser Add a new user to LDAP.

Add a new user to the LDAP directory. Specify the DN (Distinguished Name) where the new user will be created and provide user attributes as a JSON object. The request body should contain the

newUser object with user attributes and the **userDN** string that defines the DN of the new user.

Responses

Code	Description	Links
200	New user added to LDAP successfully.	No Links
400	Bad request. The newUser or userDN parameter is missing or invalid.	No Links
500	An error occurred while adding the new user to LDAP.	No Links

GET /api/v1/logs Get log entries.

Retrieve log entries based on query parameters.

Parameters

Type	Name	Description	Schema
query	period <i>optional</i>	Filter log entries by period (daily, weekly, monthly).	string
query	level <i>optional</i>	Filter log entries by level.	string

Responses

Code	Description	Links
200	A list of log entries.	No Links
401	Unauthorized or insufficient permissions.	No Links

Type	Name	Scopes
apiKey	BearerAuth	

GET /api/v1/log-file Download log file.

Download the log file containing all log entries.

Responses

Code	Description	Links
200	Log file download.	No Links
401	Unauthorized or insufficient permissions.	No Links

Type	Name	Scopes
apiKey	BearerAuth	

***POST* /api/v1/forgot-password Request a password reset.**

Request a password reset for a user.

Responses

Code	Description	Links
200	Password reset email sent successfully.	No Links
400	Bad request. Invalid input data.	No Links
404	User not found.	No Links
500	Internal Server Error.	No Links

***POST* /reset-password Restablecer la contraseña del usuario.**

Restablece la contraseña del usuario usando un código de recuperación.

Responses

Code	Description	Links
200	Contraseña restablecida exitosamente.	No Links
400	Código de recuperación inválido o caducado.	No Links
401	No autorizado. El token no es válido.	No Links
500	Error interno del servidor.	No Links

Type	Name	Scopes
apiKey	BearerAuth	

***POST* /api/v1/update-password Update user password**

Update a user's password. The user must provide the old password, a new password, and confirm the new password.

Responses

Code	Description	Links
200	Password updated successfully.	No Links
400	Bad Request. The request is missing required fields or the passwords do not match.	No Links
401	Unauthorized. The user is not authenticated.	No Links
404	Not Found. User not found.	No Links

Code	Description	Links
500	Internal Server Error.	No Links

Type	Name	Scopes
apiKey	BearerAuth	

***POST* /api/v1/login** User login.

Authenticate a user using LDAP credentials.

Responses

Code	Description	Links
200	User authenticated successfully.	No Links
401	Authentication failed. Invalid credentials or user not found.	No Links
500	An error occurred during authentication.	No Links

***POST* /api/v1/logout** User logout.

Log out a user and invalidate their access token.

Responses

Code	Description	Links
200	User logged out successfully.	No Links
400	Bad request. Access token is required.	No Links

***POST* /api/v1/refresh** Refresh access token.

Refresh the user's access token using a refresh token.

Responses

Code	Description	Links									
200	<p>Access token successfully refreshed.</p> <p><i>Content</i></p> <p>application/json</p> <p><i>Properties</i></p> <table> <tr> <th>Name</th><th>Description</th><th>Schema</th></tr> <tr> <td>newToken <i>optional</i></td><td>The new access token.</td><td>string</td></tr> <tr> <td>newRefreshToken <i>optional</i></td><td>The new refresh token.</td><td>string</td></tr> </table>	Name	Description	Schema	newToken <i>optional</i>	The new access token.	string	newRefreshToken <i>optional</i>	The new refresh token.	string	No Links
Name	Description	Schema									
newToken <i>optional</i>	The new access token.	string									
newRefreshToken <i>optional</i>	The new refresh token.	string									
401	User not found or refresh token not found.	No Links									

Components

Schemas

User

Properties

Name	Description	Schema
CI <i>optional</i>	Carnet de identidad del usuario	string
middleName <i>optional</i>	Segundo nombre del usuario	string
lastName <i>optional</i>	Apellido del usuario	string
name <i>optional</i>	Nombre del usuario	string
homeAddress <i>optional</i>	Dirección del usuario	string
telephoneNumber <i>optional</i>	Número de teléfono del usuario	string
dayRegister <i>optional</i>	Fecha de registro del usuario	string (date-time)
sex <i>optional</i>	Sexo del usuario	enum (M,F)
area <i>optional</i>	Área del usuario	string
	<i>nullable</i>	

Name	Description	Schema
userCondition <i>optional</i>	Condición del usuario	string
userStatus <i>optional</i>	Estado del usuario	enum (Activo, Inactivo)
sedeMunicipio <i>optional</i>	Municipio de la sede del usuario	string
userType <i>optional</i>	Tipo de usuario	enum (Estudiante, Trabajador, Trabajador Docente)
userInfo <i>optional</i>	Información del usuario acerca del curso	string
career <i>optional</i>	Carrera del usuario	string
studentClassGroup <i>optional</i>	Grupo del usuario	string
studentYear <i>optional</i>	Año del usuario	string
country <i>optional</i>	País del usuario Default: Cuba	string
UJC <i>optional</i>	Pertenece a la UJC	string
skinColor <i>optional</i>	Color de piel del usuario	string
nameInstitution <i>optional</i>	Nombre de la institución del usuario Default: CUJAE	string
right <i>optional</i>	Derecho del usuario Default: Todos	string
hash <i>optional</i>	Hash de la contraseña del usuario	string
lastTimeChange <i>optional</i>	Fecha de la última actualización de la contraseña	string (date-time)
uid <i>optional</i>	UID del usuario. Formato: nombre.apellido	string
homeDirectory <i>optional</i>	Directorio del usuario. Formato: /home/uid	string
givenName <i>optional</i>	Nombre del usuario	string
cn <i>optional</i>	Nombre completo del usuario	string

Name	Description	Schema
sn <i>optional</i>	Apellido del usuario	string
displayName <i>optional</i>	Nombre de visualización del usuario	string
uidNumber <i>optional</i>	Número de UID del usuario. Formato: 1000	string
userPassword <i>optional</i>	Contraseña del usuario. Formato: {SSHA}hash	string
mail <i>optional</i>	Correo electrónico del usuario	< string > array
maildrop <i>optional</i>	Correo electrónico del usuario	< string > array
gidNumber <i>optional</i>	Número de GID del usuario. Formato: 1000	string
sambaSID <i>optional</i>	SID del usuario. Formato: S-1-5-21-1004336348-1177238915-682003330-1000	string
objectClass <i>optional</i>	Clases del usuario. Formato: top, person, organizationalPerson, inetOrgPerson, posixAccount, shadowAccount, sambaSamAccount	< string > array

Student

Properties

Name	Description	Schema
career <i>optional</i>		string
studentYear <i>optional</i>		string
studentClassGroup <i>optional</i>		string
userInformation <i>optional</i>		string
userCondition <i>optional</i>		string
userStatus <i>optional</i>		string

Employee

Properties

Name	Description	Schema
dateContract <i>optional</i>		string (date-time)
dateEndContract <i>optional</i>		string (date-time)
educationalCategory <i>optional</i>		string
orgRole <i>optional</i>		string
schoolLevel <i>optional</i>		string
scientificCategory <i>optional</i>		string
userYears <i>optional</i>		string
workerContract <i>optional</i>		string
workArea <i>optional</i>		string
workerID <i>optional</i>		string

LdapEntry

JWTPayload

Properties

Name	Description	Schema
sub <i>optional</i>	Subject (sub) del usuario.	string
dn <i>optional</i>	Distinguished Name (DN) del usuario.	string
uid <i>optional</i>	Identificador único del usuario.	string
groups <i>optional</i>	Grupos a los que pertenece el usuario.	< string > array
base <i>optional</i>	Base DN principal.	string
localBase <i>optional</i>	Base DN local.	string
firstname <i>optional</i>	Nombre del usuario.	string
lastname <i>optional</i>	Apellido del usuario.	string

Name	Description	Schema
fullname <i>optional</i>	Nombre completo del usuario.	string
email <i>optional</i>	Dirección de correo electrónico del usuario.	string (email)
ci <i>optional</i>	Número de identidad del usuario.	string
roles <i>optional</i>	Roles del usuario.	< string > array
last_time_logged <i>optional</i>	Fecha y hora de la última sesión iniciada.	string (date-time)
loginInfo <i>optional</i>	Información de inicio de sesión.	string
iat <i>optional</i>	Tiempo de emisión del token JWT.	integer (int64)
exp <i>optional</i>	Tiempo de expiración del token JWT.	integer (int64)

UserResponse

Properties

Name	Description	Schema
status <i>optional</i>	El estado de la respuesta	string
data <i>optional</i>		object