## Create Custom Endpoint

Create Custom Endpoint to redirect its requests to another endpoints

|  |  |
| --- | --- |
| **Path** | /admin/customendpoints |
| **Method** | **POST** |

### Request

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **Name** | **Description** | **Schema/Value** |
| HEADERS | Content-Type | Content Type of Request | application/json |
| BODY | request | CustomEndpoint payload | handlers.SwaggerCustomEndpoint [object] |

### Response

Content-Type: application/json

|  |  |  |
| --- | --- | --- |
| **HTTP Code** | **Description** | **Schema** |
| **200** | **JWT token** | **No Content** |
| 400 | Invalid JSON format | No Content |
| 401 | Unauthorized: invalid credentials | No Content |
| 500 | Server error during token generation | No Content |

### Sample Request [body]

{  
 "method": "\"\"",  
 "needAccounting": "\"\"",  
 "path": "\"\""  
}

## Get admin dashboard data

Retrieves a list of all users (username + role) and all accounting rules. Admins only.

|  |  |
| --- | --- |
| **Path** | /admin/dashboard |
| **Method** | **GET** |

### Request

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **Name** | **Description** | **Schema/Value** |

### Response

Content-Type: application/json

|  |  |  |
| --- | --- | --- |
| **HTTP Code** | **Description** | **Schema** |
| **200** | **OK** | **handlers.AdminDashboardResponse [object]** |
| 401 | Token claims missing or invalid | No Content |
| 403 | Not an admin | No Content |
| 500 | Database error | No Content |

### Sample Response [body]

HTTP Code: 200

{  
 "message": "\"\"",  
 "rules": [  
 {  
 "charge": "\"\"",  
 "created\_at": "\"\"",  
 "endpoint": "\"\"",  
 "id": "\"\"",  
 "updated\_at": "\"\""  
 }  
 ],  
 "users": [  
 {  
 "role": "\"\"",  
 "username": "\"\""  
 }  
 ]  
}

## Login a user

Authenticate user credentials and return a signed JWT

|  |  |
| --- | --- |
| **Path** | /login |
| **Method** | **POST** |

### Request

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **Name** | **Description** | **Schema/Value** |
| HEADERS | Content-Type | Content Type of Request | application/json |
| BODY | request | Login payload | handlers.LoginRequest [object] |

### Response

Content-Type: application/json

|  |  |  |
| --- | --- | --- |
| **HTTP Code** | **Description** | **Schema** |
| **200** | **JWT token** | **No Content** |
| 400 | Invalid JSON format | No Content |
| 401 | Unauthorized: invalid credentials | No Content |
| 500 | Server error during token generation | No Content |

### Sample Request [body]

{  
 "captchaId": "\"\"",  
 "captchaSolution": "\"\"",  
 "password": "\"\"",  
 "username": "\"\""  
}

## List all roles

Returns a list of all roles defined in the system.

|  |  |
| --- | --- |
| **Path** | /roles |
| **Method** | **GET** |

### Response

Content-Type: application/json

|  |  |  |
| --- | --- | --- |
| **HTTP Code** | **Description** | **Schema** |
| **200** | **OK** | **handlers.RolesListResponse [object]** |
| 500 | Internal Server Error | handlers.ErrorResponse [object] |

### Sample Response [body]

HTTP Code: 200

{  
 "roles": [  
 {  
 "description": "\"\"",  
 "id": "\"\"",  
 "name": "\"\""  
 }  
 ]  
}

HTTP Code: 500

{  
 "details": "\"\"",  
 "error": "\"\""  
}

## Create a new role

Allows an admin to create a new role by specifying name and description.

|  |  |
| --- | --- |
| **Path** | /roles |
| **Method** | **POST** |

### Request

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **Name** | **Description** | **Schema/Value** |
| HEADERS | Content-Type | Content Type of Request | application/json |
| BODY | role | Role details | handlers.SwaggerRoleRequest [object] |

### Response

Content-Type: application/json

|  |  |  |
| --- | --- | --- |
| **HTTP Code** | **Description** | **Schema** |
| **200** | **OK** | **handlers.SuccessResponse [object]** |
| 400 | Bad Request | handlers.ErrorResponse [object] |
| 500 | Internal Server Error | handlers.ErrorResponse [object] |

### Sample Request [body]

{  
 "description": "\"\"",  
 "name": "\"\""  
}

### Sample Response [body]

HTTP Code: 200

{  
 "message": "\"\"",  
 "role": {  
 "description": "\"\"",  
 "id": "\"\"",  
 "name": "\"\""  
 }  
}

HTTP Code: 400

{  
 "details": "\"\"",  
 "error": "\"\""  
}

HTTP Code: 500

{  
 "details": "\"\"",  
 "error": "\"\""  
}

## Register a new user

Create a new user account with username, password, and role

|  |  |
| --- | --- |
| **Path** | /users |
| **Method** | **POST** |

### Request

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **Name** | **Description** | **Schema/Value** |
| HEADERS | Content-Type | Content Type of Request | application/json |
| BODY | request | Registration payload | handlers.RegisterRequest [object] |

### Response

Content-Type: application/json

|  |  |  |
| --- | --- | --- |
| **HTTP Code** | **Description** | **Schema** |
| **200** | **User registered successfully** | **No Content** |
| 400 | Invalid input or missing fields | No Content |
| 500 | Server error during registration | No Content |

### Sample Request [body]

{  
 "password": "\"\"",  
 "role": "\"\"",  
 "username": "\"\""  
}

## Delete a user

Delete an existing user account (admin only)

|  |  |
| --- | --- |
| **Path** | /users/{username} |
| **Method** | **DELETE** |

### Request

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **Name** | **Description** | **Schema/Value** |
| PATH | username | Username to delete | string |

### Response

Content-Type: application/json

|  |  |  |
| --- | --- | --- |
| **HTTP Code** | **Description** | **Schema** |
| **200** | **User deleted successfully** | **No Content** |
| 400 | Username is required | No Content |
| 404 | User not found | No Content |
| 500 | Could not delete user | No Content |

## Update user role

Update the role of an existing user (admin only)

|  |  |
| --- | --- |
| **Path** | /users/{username}/role |
| **Method** | **PUT** |

### Request

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **Name** | **Description** | **Schema/Value** |
| HEADERS | Content-Type | Content Type of Request | application/json |
| PATH | username | Username to update | string |
| BODY | request | Role update payload | handlers.RoleUpdateRequest [object] |

### Response

Content-Type: application/json

|  |  |  |
| --- | --- | --- |
| **HTTP Code** | **Description** | **Schema** |
| **200** | **User role updated successfully** | **No Content** |
| 400 | Invalid input or missing fields | No Content |
| 404 | User not found | No Content |
| 500 | Failed to update user role | No Content |

### Sample Request [body]

{  
 "role": "\"\""  
}

## Models

### handlers.AdminDashboardResponse

string **message**  
handlers.SwaggerAccountingRule **rules**  
handlers.SwaggerUserRolePair **users**

### handlers.ErrorResponse

string **details**  
string **error**

### handlers.LoginRequest

string **captchaId**  
string **captchaSolution**  
string **password**  
string **username**

### handlers.RegisterRequest

string **password**  
string **role**  
string **username**

### handlers.RoleUpdateRequest

string **role**

### handlers.RolesListResponse

handlers.SwaggerRole **roles**

### handlers.SuccessResponse

string **message**  
handlers.SwaggerRole **role**

### handlers.SwaggerAccountingRule

number **charge**  
string **created\_at**  
string **endpoint**  
integer **id**  
string **updated\_at**

### handlers.SwaggerCustomEndpoint

string **endpoints**  
string **method**  
boolean **needAccounting**  
string **path**

### handlers.SwaggerRole

string **description**  
integer **id**  
string **name**

### handlers.SwaggerRoleRequest

string **description**  
string **name**

### handlers.SwaggerUserRolePair

string **role**  
string **username**