

# API Documentation - DFCU Staff Management System

## Base URL

Base URL for staff requests: <http://localhost:3000/api/staff>

Base URL for admin requests: <http://localhost:3000/api/admin>

## 1. Register a New Staff Member

- **Endpoint:** /register
- **Method:** POST
- **Description:** Registers a new staff member, including optional file upload for an ID photo.

### Request

- **Headers:** `Content-Type: multipart/form-data`

### Form Data

- **surname:** (string) The staff member's surname. *(Required)*
- **otherNames:** (string) The staff member's other names. *(Required)*
- **dateOfBirth:** (string, format: YYYY-MM-DD) The staff member's date of birth. *(Required)*
- **authCode:** (string) Authorization code for the staff member. *(Required, length: 10 digits)*
- **idPhoto:** (file) An optional image of the staff member's ID photo.

### Example Request

```
POST /api/staff/register HTTP/1.1
Host: localhost:3000
Content-Type: multipart/form-data
{
  "surname": "Doe",
  "otherNames": "John",
  "dateOfBirth": "1980-05-12",
  "authCode": "1234567890",
  "idPhoto": (file)
}
```

### Example Response (Success)

```
{
  "message": "Staff registered successfully.",
  "employeeNumber": "XYZ12345"
```

```
}
```

### Example Response (Error)

```
{
  "message": "Registration Failed",
  "errors": [
    "Invalid auth code."
  ]
}
```

## 2. Retrieve Staff Members

- **Endpoint:** /retrieve
- **Method:** GET
- **Description:** Retrieve one staff member by employee number or all staff members if no number is provided.

### Request:

- **Query Parameters:** employeeNumber (optional): (string) Employee number to retrieve a specific staff member.

### Example Request

```
GET /api/staff/retrieve?employeeNumber=XYZ12345 HTTP/1.1
Host: localhost:3000
```

### Example Response (Success)

```
{
  "employeeNumber": "XYZ12345",
  "surname": "Doe",
  "otherNames": "John",
  "dateOfBirth": "1980-05-12",
  "idPhoto": "base64-encoded-image-data",
  "authCode": "1234567890"
}
```

### Example Response (Error)

```
{
  "message": " Failed to retrieve staff member(s).",
}
```

## 3. Update Staff Member Details

- **Endpoint:** /update/:employeeNumber
- **Method:** PUT

- **Description:** Update a staff member's details, including date of birth and ID photo.

#### Request:

- **URL Parameters:** employeeNumber: (string) The unique employee number of the staff member to update.
- **Headers:** `Content-Type: multipart/form-data`

Form Data:

- **dateOfBirth:** (string, format: YYYY-MM-DD) *(Optional)*
- **idPhoto:** (file) *(Optional)*

#### Example Request

```
PUT /api/staff/update/XYZ12345 HTTP/1.1
Host: localhost:3000
Content-Type: multipart/form-data
{
  "dateOfBirth": "1990-01-15",
  "idPhoto": (file)
}
```

#### Example Response (Success)

```
{
  "message": "Staff member updated successfully."
}
```

#### Example Response (Success)

```
{
  "message": "Staff member not found."
}
```

## 4. Admin Registration

- **Endpoint:** /register
- **Method:** POST
- **Description:** Registers a new admin using their employee number. A password is auto-generated.

#### Request:

- **Body:**
  - employeeNumber: (string) The employee number of the admin.

#### Example Request:

```
{
  "message": "Admin registered successfully.",
}
```

```
"employeeNumber": "ADMIN123",  
"generatedPassword": "abc12345"  
}
```

#### Example Response (Success)

```
{  
  "message": "Admin registered successfully.",  
  "employeeNumber": "ADMIN123",  
  "generatedPassword": "abc12345"  
}
```

#### Example Response (Error)

```
{  
  "message": "Admin with this employee number already exists."  
}
```

## 5. Admin Login

- **Endpoint:** /login
- **Method:** POST
- **Description:** Logs in an admin using their employee number and password, and returns a JWT token.

#### Request:

- **Body:**
  - **employeeNumber:** (string) The employee number of the admin.
  - **password:** (string) The admin's password.

#### Example Request

```
POST /api/admin/login HTTP/1.1  
Host: localhost:3000  
Content-Type: application/json  
{  
  "employeeNumber": "ADMIN123",  
  "password": "abc12345"  
}
```

#### Example Response (Success)

```
{  
  "message": "Login successful.",  
  "token": "jwt-token"  
}
```

#### Example Response (Error)

```
{
  "message": "Invalid employee number or password."
}
```

## 6. Fetch Admin Metrics (Protected Route)

- **Endpoint:** /metrics
- **Method:** GET
- **Description:** Fetches admin metrics like the number of requests, successful requests, and failed requests.

### Request:

- **Headers:** `Authorization: Bearer jwt-token`

### Example Request:

```
GET /api/admin/metrics HTTP/1.1
Host: localhost:3000
Authorization: Bearer jwt-token
```

### Example Response (Success):

```
{
  "totalRequests": 50,
  "successfulRequests": 45,
  "failedRequests": 5
}
```

### Example Response (Error):

```
{
  "message": "Failed to retrieve"
}
```

## General Errors:

For all endpoints, common errors include:

### 1. 500 Internal Server Error:

```
{
  "message": "An internal server error occurred."
}
```

### 2. 404 Not Found (e.g., staff or admin not found):

```
{
  "message": "Resource not found."
}
```

### 400 Bad Request (e.g., validation errors):

```
{
  "message": "Invalid request data.",
  "errors": [
    "Employee number is required",
    "Password must be at least 6 characters long"
  ]
}
```