

# DOCUMENTATION

Below are instructions for setting up and running the project along with details on API endpoints, user registration, login, and authentication token usage. Please note that this is a simplified example, and in a real-world scenario, you might want to add more security measures, error handling, and thorough testing.

## Setting up and Running the Project:

- Clone the repository:

bashCopy code

**clone**

- Install dependencies:

bashCopy code

- Apply migrations:

bashCopy code

- Create a superuser (admin):

bashCopy code

Follow the prompts to create a superuser. Remember the credentials; you'll use them to access the Django admin interface.

- Run the development server:

bashCopy code

Visit <http://localhost:8000/admin/> and log in with the superuser credentials to access the Django admin interface.

## API Endpoints and Usage:

### 1. User Registration:

- **Endpoint:** <http://localhost:8000/api/register/>
- **Method:** POST
- **Headers:**
  - **Content-Type:** application/json
- **Body:**

jsonCopy code

```
"username" "your_username"
"password" "your_password"
}
```

- **Note:** This endpoint will create a new user and automatically generate an associated artist object using signals.

## 2. User Login and Token Retrieval:

- **Endpoint:** <http://localhost:8000/api/token/>
- **Method:** POST
- **Headers:**
  - **Content-Type:** application/json
- **Body:**

jsonCopy code

```
"username" "your_username"
"password" "your_password"
}
```

- **Response:**

jsonCopy code

```
"access" "your_access_token"
"refresh" "your_refresh_token"
}
```

- **Note:** You will use the **access** token for authentication in subsequent requests.

## 3. List of Works:

- **Endpoint:** <http://localhost:8000/api/works/>
- **Method:** GET
- **Headers:**
  - **Authorization:** Bearer your\_access\_token

## 4. Create a New Work:

- **Endpoint:** <http://localhost:8000/api/works/>
- **Method:** POST
- **Headers:**
  - **Authorization:** Bearer your\_access\_token
  - **Content-Type:** application/json
- **Body:**

jsonCopy code

```
"link" "https://example.com"
"work_type" "YT"
}
```

#### 5. Filter Works by Work Type:

- **Endpoint:** [http://localhost:8000/api/works?work\\_type=YT/](http://localhost:8000/api/works?work_type=YT/)
- **Method:** GET
- **Headers:**
  - **Authorization:** Bearer your\_access\_token

#### 6. Search for Artists by Name:

- **Endpoint:** [http://localhost:8000/api/works/search/your\\_artist\\_name/](http://localhost:8000/api/works/search/your_artist_name/)
- **Method:** GET
- **Headers:**
  - **Authorization:** Bearer your\_access\_token

### Dummy Data:

For testing purposes, you can use the Django admin interface to manually add dummy data:

- Visit <http://localhost:8000/admin/> and log in with the superuser credentials.
- Navigate to "Artists" and "Works" sections to add artists and works manually.