

Online Auction Platform

Introduction

The **Online Auction Platform** is a Django-based application designed to enable users to list, bid on, and manage auction items. This platform supports multiple user roles, including Admin, Seller, Buyer, and Moderator, each with specific permissions and capabilities.

Features

- User registration and authentication (JWT-based).
- Role-based access control (Admin, Seller, Buyer, Moderator).
- Create, update, and delete auction items (Seller).
- Place bids on auction items (Buyer).
- Moderate auction content (Moderator).
- Admin panel for managing users and platform settings.

Technologies Used

- **Backend:** Django, Django REST Framework (DRF), Simple JWT for authentication.
- **Frontend:** Postman for API testing.
- **Database:** PostgreSQL (can be replaced with SQLite/MySQL).
- **Other:** Python 3.12, Virtual Environment for project isolation.

Installation Instructions

1. Clone the repository:
2. Create and activate a virtual environment:

```
python3 -m venv env
source env/bin/activate
```

3. Install the required dependencies:

```
pip install -r requirements.txt
```

4. Run migrations:

```
python manage.py makemigrations
python manage.py migrate
```

5. Create a superuser:

```
python manage.py createsuperuser
```

6. Start the development server:

```
python manage.py runserver
```

User Roles

Admin

- Manage all users and roles.
- Full access to all auction items and bids.

Seller

- List items for auction.
- Update or delete their own auction items.

Buyer

- Browse auction items.
- Place bids on items.

Moderator

- Review and moderate auction content.
- Remove inappropriate listings.

Testing with Postman

1. Import the API endpoints into Postman.
2. Use the `/api/auth/token/` endpoint to obtain a JWT token.
3. Add the token to the Authorization header for subsequent requests:

```
Authorization: Bearer <token>
```

License

This project is licensed under the MIT License. See the LICENSE file for details.