

Real Estate App - Mini API Documentation

Mini API Documentation

Base URL:

http://localhost:5000/api (Use production URL when deployed)

Authentication:

JWT-based. Include this in headers for all protected routes:

Authorization: Bearer <token>

AUTH ROUTES (/auth)

POST /auth/register

- Register a new user (buyer or agent)

- Body:

```
{
  "username": "john_doe",
  "email": "john@example.com",
  "password": "securepassword",
  "role": "agent" // or "buyer"
}
```

POST /auth/login

- Log in a user

- Body:

```
{
  "email": "john@example.com",
  "password": "securepassword"
}
```

USER ROUTES (/users)

GET /users/:id

- Fetch a user by ID

PATCH /users/:id

- Update user profile

PROPERTY ROUTES (/property)

All routes require Authorization header.

POST /property/create

- Create a new property (agent-only)

Real Estate App - Mini API Documentation

- Body includes title, description, price, location, images

GET /property/realtor/

- Get properties listed by logged-in agent

GET /property/all

- Get all properties

GET /property/search?location=&priceMin=&priceMax=

- Search with filters

PUT /property/:id

- Update a property by ID

DELETE /property/:id

- Delete a property by ID

POST ROUTES (/posts)

GET /posts

- Fetch all posts

POST /posts

- Create a new post

DOWNLOAD BADGE (/downloadBadge)

GET /downloadBadge/:userId

- Downloads badge/certificate for a user

WAITLIST ROUTES (/waitlist)

POST /waitlist

- Add email/user to waitlist

- Body:

```
{  
  "email": "example@example.com"  
}
```

IMAGE UPLOAD

Static files hosted at:

Real Estate App - Mini API Documentation

GET /images/<filename>

e.g. <http://localhost:5000/images/house1.jpg>

USER ROLES

- agent: Can post and manage properties
- buyer: Can browse properties and send interest
- admin: Full access (if implemented)

ERROR CODES

401 Unauthorized Missing/invalid token

403 Forbidden Insufficient permissions

404 Not Found Resource doesn't exist

500 Internal Server Error Something broke