

# Home Service Platform – API Documentation

## Base URL

http://<domain>:3000/api

## Authentication APIs

| Method | Endpoint       | Auth Needed | Description             |
|--------|----------------|-------------|-------------------------|
| POST   | /auth/register | No          | Register a new user     |
| POST   | /auth/login    | No          | Login and get JWT token |

### Register

POST /auth/register

#### Request Body:

```
{
  "name": "John Doe",
  "email": "john@example.com",
  "mobile": "9876543210",
  "password": "mypassword",
  "role": "customer"
}
```

#### Success Response (201):

```
{
  "success": true,
  "message": "User registered successfully",
  "data": {
    "user": { ... },
    "token": "..."
  }
}
```

### Login

POST /auth/login

### Request Body:

```
{
  "emailOrMobile": "john@example.com",
  "password": "mypassword"
}
```

### Success Response (200):

```
{
  "success": true,
  "message": "Login successful",
  "data": {
    "user": { ... },
    "token": "... "
  }
}
```

## Profile APIs

| Method | Endpoint | Auth Needed    | Description                |
|--------|----------|----------------|----------------------------|
| GET    | /me      | JWT (any role) | Get logged-in user profile |
| PUT    | /me      | JWT (any role) | Update profile             |

## Address APIs

| Method | Endpoint       | Auth Needed       | Description                |
|--------|----------------|-------------------|----------------------------|
| POST   | /addresses     | Customer/Provider | Create new address         |
| GET    | /addresses     | Customer/Provider | Get current user addresses |
| PUT    | /addresses/:id | Customer/Provider | Update an address          |
| DELETE | /addresses/:id | Customer/Provider | Delete an address          |

### Create Address

**POST** /addresses

```
{
  "label": "Home",
  "line1": "123 Main St",
}
```

```

"line2": "Near Park",
"city": "Ahmedabad",
"state": "Gujarat",
"pincode": "380001",
"latitude": 23.033863,
"longitude": 72.585022
}

```

## Service Categories and Services

| Method | Endpoint                 | Auth Needed | Description               |
|--------|--------------------------|-------------|---------------------------|
| GET    | /categories              | No          | List service categories   |
| GET    | /categories/:id/services | No          | List services by category |
| GET    | /services                | No          | List all services         |
| GET    | /services/:id            | No          | Get service details       |

## Booking APIs

| Method | Endpoint      | Auth Needed       | Description                        |
|--------|---------------|-------------------|------------------------------------|
| POST   | /bookings     | Customer          | Create a new booking               |
| GET    | /bookings     | Customer/Provider | Get all bookings for user/provider |
| GET    | /bookings/:id | Customer/Provider | Get booking details                |
| PUT    | /bookings/:id | Provider          | Update booking status or assign    |

## Payment APIs

| Method | Endpoint             | Auth Needed       | Description               |
|--------|----------------------|-------------------|---------------------------|
| POST   | /payments            | Customer          | Create a payment entry    |
| GET    | /payments/:bookingId | Customer/Provider | Get payment for a booking |

## Review APIs

| Method | Endpoint             | Auth Needed | Description                      |
|--------|----------------------|-------------|----------------------------------|
| POST   | /reviews             | Customer    | Add review for completed booking |
| GET    | /reviews/:providerId | No          | List reviews for a provider      |

## Notification APIs

| Method | Endpoint                        | Auth Needed | Description                         |
|--------|---------------------------------|-------------|-------------------------------------|
| GET    | <code>/notifications</code>     | Any role    | List logged-in user's notifications |
| PUT    | <code>/notifications/:id</code> | Any role    | Mark notification as read           |

## Provider Earnings and Payout APIs

| Method | Endpoint                                | Auth Needed | Description               |
|--------|---|-------------|---------------------------|
| GET    | <code>/providers/earnings</code>        | Provider    | Get earnings summary      |
| POST   | <code>/providers/payout-requests</code> | Provider    | Request payout from admin |
| GET    | <code>/providers/payout-requests</code> | Provider    | List my payout requests   |

## Admin APIs

| Method | Endpoint                      | Auth Needed | Description                        |
|--------|-------------------------------|-------------|------------------------------------|
| GET    | <code>/admin/logs</code>      | Admin       | Get list of admin logs and actions |
| GET    | <code>/admin/users</code>     | Admin       | List all users                     |
| PATCH  | <code>/admin/users/:id</code> | Admin       | Update user status or role         |

## Testing

### Check if the API is up:

```
GET http://localhost:3000/api/test
```

### Response:

```
{
  "success": true,
  "message": "Test route works!"
}
```

## Authorization

All routes that require authentication expect:

Authorization: Bearer <your\_jwt\_token\_here>

## Notes

- All timestamps use UTC
- Default pagination limit (if added later): limit=10
- Error responses follow structure:

```
{  
  "success": false,  
  "message": "Error message here",  
  "error": "Optional developer error message or stack"  
}
```

## Future Upgrades

- Password reset endpoints
- Multi-language support
- Google OAuth login
- Webhooks for payment providers
- Rate limit control