API Documentation

1) REGISTER

Endpoint: POST /api/auth/register

Request :

```
{
  "email": "Abhay@gmail.com",
  "password": "securepasswords123",
  "organization": "TechMam"
}
```

Response:

```
{
  "id": "693acf37898cc68ec9580d08",
  "email": "Abhay@gmail.com",
  "role": "user"
}
```

Request :

```
{
  "email": "shivam@gmail.com",
  "password": "securepasswords123",
  "organization": "Techdom"
}
```

Response:

```
{
  "id": "693acf9a898cc68ec9580d10",
  "email": "shivam@gmail.com",
  "role": "user"
}
```

2) LOGIN

Endpoint: POST /api/auth/login

Request:

```
{
  "email": "shivam@gmail.com",
  "password": "securepasswords123",
  "organization": "Techdom"
}
```

Response :

```
{
  "token":
"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VySWQiOiI2OTNhY2Y5YTg5OGNjNjhlYzk1
ODBkMTAiLCJyb2xlIjoidXNlciIsIm9yZ2FuaXphdGlvbklkIjoiNjkzYWNmOTk4OThjYzY4ZWM
5NTgwZDBkIiwiaWF0IjoxNzY1NDYyMDE1LCJleHAiOjE3NjU0OTA4MTV9.pp1g_ke0rmjc30
4Gq13pPZvuPMx44GaU914bgZIdrNI",
  "user": {
    "id": "693acf9a898cc68ec9580d10",
    "email": "shivam@gmail.com",
    "role": "user",
    "organizationId": "693acf99898cc68ec9580d0d"
  }
}
```

3) CREATE USER (Admin Only) - using provided token

Endpoint: POST /api/users

Headers:

```
Authorization: Bearer
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VySWQiOiI2OTNhY2Y5YTg5OGNjNjhlYzk1O
DBkMTAiLCJyb2xlIjoidXNlciIsIm9yZ2FuaXphdGlvbklkIjoiNjkzYWNmOTk4OThjYzY4ZWM5
NTgwZDBkIiwiaWF0IjoxNzY1NDYyMDE1LCJleHAiOjE3NjU0OTA4MTV9.pp1g_ke0rmjc304
Gq13pPZvuPMx44GaU914bgZIdrNI
```

```
Content-Type: application/json
```

Request Body:

```
{
  "email": "employee@example.com",
  "password": "employee123",
  "role": "user"
}
```

Response (example success):

```
{
  "id": "693ad10e898cc68ec9580d55",
  "email": "employee@example.com",
  "role": "user",
  "organizationId": "693acf99898cc68ec9580d0d"
}
```

4) UPDATE USER ROLE - using provided token

Endpoint: PUT /api/users/:USER_ID

Example URL: PUT /api/users/693ad10e898cc68ec9580d55

Headers:

Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VySWQiOiI2OTNhY2Y5YTg5OGNjNjhlYzk1ODBkMTAiLCJyb2xlIjoidXNlciIsIm9yZ2FuaXphdGlvbklkIjoiNjkzYWNmOTk4OThjYzY4ZWM5NTgwZDBkIiwiaWF0IjoxNzY1NDYyMDE1LCJleHAiOjE3NjU0OTA4MTV9.pp1g_ke0rmjc304Gq13pPZvuPMx44GaU914bgZIdrNI

Content-Type: application/json

Request Body:

```
{
  "role": "manager"
}
```

Success Response:

```
{
  "id": "693ad10e898cc68ec9580d55",
  "email": "employee@example.com",
  "role": "manager",
  "organizationId": "693acf99898cc68ec9580d0d"
}
```

Errors (examples):

{ "message": "Not found" }
{ "message": "Forbidden" }
{ "message": "Invalid token" }
{ "message": "User exists" }

5) DELETE USER - using provided token

Endpoint: DELETE /api/users/:USER_ID

Example URL: DELETE /api/users/693ad10e898cc68ec9580d55

Headers:

Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VySWQiOiI2OTNhY2Y5YTg5OGNjNjhlYzk1ODBkMTAiLCJyb2xlIjoidXNlciIsIm9yZ2FuaXphdGlvbklkIjoiNjkzYWNmOTk4OThjYzY4ZWM5NTgwZDBkIiwiaWF0IjoxNzY1NDYyMDE1LCJleHAiOjE3NjU0OTA4MTV9.pp1g_ke0rmjc304Gq13pPZvuPMx44GaU914bgZIdrNI

Success Response:

```
{
  "message": "Deleted"
}
```

Errors (examples):

{ "message": "Not found" }
{ "message": "Forbidden" }
{ "message": "Invalid token" }

⭐ 6) PROTECTED ROUTE EXAMPLES (Using JWT Token)

All protected endpoints require the Authorization header:

Authorization: Bearer <YOUR_JWT_TOKEN>

Example Token (from login screenshot):

Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VySWQiOiI2OTNhY2Y5YTg5OGNjNjhlYzk1ODBkMTAiLCJyb2xlIjoidXNlciIsIm9yZ2FuaXphdGlvbklkIjoiNjkzYWNmOTk4OThjYzY4ZWM5NTgwZDBkIiwiaWF0IjoxNzY1NDYyMDE1LCJleHAiOjE3NjU0OTA4MTV9.pp1g_ke0rmjc304Gq13pPZvuPMx44GaU914bgZIdrNI

Example: GET /api/users (Admin or Manager)

Headers:

Authorization: Bearer <TOKEN>

Response Example:

```
{
  "users": [
    {
      "id": "693acf9a898cc68ec9580d10",
      "email": "shivam@gmail.com",
      "role": "user",
      "organizationId": "693acf99898cc68ec9580d0d"
    }
  ]
}
```

⭐ 7) EXTERNAL API KEY USAGE EXAMPLES

External systems can interact using API keys instead of JWT.

API Key must be sent via:

x-api-key: <YOUR_API_KEY>

```
------------------------------------

Example: API Key Creation (Admin Only)

POST /api/apikeys

------------------------------------

Response:

{
  "apiKeyId": "key_001",
  "apiKey": "generated_api_key_value"
}


------------------------------------

Example: API Key Rotation

POST /api/apikeys/rotate/key_001

------------------------------------

Response:

{
  "apiKeyId": "key_002",
  "apiKey": "new_rotated_api_key"
}


------------------------------------

Example: API Key Revocation

POST /api/apikeys/revoke/key_002

------------------------------------

Response:

{
  "message": "Revoked"
}
```