

User Registration API – Technical Document

Overview

The User Registration API allows new users to register by submitting their basic information. OTP verification and JWT are **not** used in this implementation. The API accepts user details, validates them, and stores them in the database.

Endpoint

- **Method:** POST
- **URL:** /register/
- **Authentication:** Not Required

Request Payload Example

```
{
  "first_name": "John",
  "last_name": "Doe",
  "date_of_birth": "1995-05-20",
  "age": 28,
  "mobile_number": "9876543210",
  "emergency_contact": "9123456789",
  "blood_group": "A+",
  "gender": "MALE",
  "instrument": "Guitar",
  "address": "123 Music Lane"
}
```

Success Response (201 Created)

```
{
  "data": {
    "id": 1,
    "first_name": "John",
    "last_name": "Doe",
    "date_of_birth": "1995-05-20",
    "age": 28,
    "mobile_number": "9876543210",
    "emergency_contact": "9123456789",
    "blood_group": "A+",

```

```
"gender": "MALE",
"instrument": "Guitar",
"address": "123 Music Lane",
"created_at": "...",
"updated_at": "...",
"soft_delete": 0,
"updated_by": null
},
"type": "success"
}
```

✖ Error Response (400 Bad Request)

```
{
  "mobile_number": ["user with this mobile number already exists."]
}
```

⚙ Key Functional Notes

- Mobile number must be unique.
- Gender uses predefined choices: "MALE", "FEMALE", "OTHER".
- Timestamps are auto-generated: `created_at`, `updated_at`.
- `soft_delete` defaults to `0` (active).
- `updated_by` is null unless linked with an authenticated user.

🧱 Technology Stack

- **Backend:** Python (Django + Django REST Framework)
- **Database:** PostgreSQL

🛡 Security Notes

- HTTPS recommended in production.
- No JWT or OTP used in current implementation.
- Use environment variables for secrets and sensitive data.

