

# Meal Bridge - Model Documentation

## 1. User Model (Abstract Base Model)

Since we have two types of users (Restaurant Owners & Donees), we use Django's `AbstractUser` to customize authentication.

### Fields:

- `email` (Primary login field)
  - `password`
  - `role` (`restaurant_owner` / `donee`)
  - `is_verified` (Boolean, defaults to `False` for donees, set to `True` after manual verification)
  - `verification_token` (For email verification)
  - `profile_picture` (ImageField, optional)
  - `phone_number`
  - `created_at` (Auto timestamp)
- 

## 2. Donee Model (For NGOs/Charitable Organizations)

### Fields:

- `user` (One-to-One relation with `User`)
- `organization_name`
- `contact_person_name`
- `phone_number`
- `location`
- `registration_certificate` (File Upload)
- `is_approved` (Boolean, manually set `True` after verification)
- `previous_donations_received` (ManyToManyField to `FoodDonation`)

### Features:

- Profile image for the NGO
  - A list of all donations they have received
- 

## 3. Restaurant Model

### Fields:

- `user` (One-to-One relation with `User`)

- `restaurant_name`
- `owner_name`
- `phone_number`
- `location`
- `cuisine_type` (ChoiceField: `Indian`, `Chinese`, `Italian`, `Mexican`, etc.)
- `restaurant_logo` (ImageField, optional)
- `food_donations` (Related to `FoodDonation`)

**Features:**

- Store cuisine type to help NGOs identify preferred food
  - Upload restaurant logo
- 

## 4. Food Donation Model

**Fields:**

- `restaurant` (ForeignKey to `Restaurant`)
- `food_details` (TextField for description)
- `quantity`
- `expiry_time` (Datetime – when the food will expire)
- `preferred_organizations` (ManyToManyField with `Donee`)
- `is_claimed` (Boolean, defaults to `False`)
- `created_at` (Auto timestamp)
- `image` (ImageField for food picture, optional)

**Features:**

- Store images of food donations
  - Store which NGO accepted the donation
- 

## 5. Notifications Model

**Fields:**

- `user` (ForeignKey to `User`, recipient of notification)
- `message`
- `is_read` (Boolean, defaults to `False`)
- `timestamp`

**Functionality:**

- NGOs get notifications when a restaurant posts a donation
- Restaurants get notified when an NGO accepts the donation

---

## 6. Email Notification for Donees

When a restaurant submits a donation, the selected donees will receive an email notification.

### Implementation using Django's Email Module

```
from django.core.mail import send_mail
from django.conf import settings

def send_email(to_email, subject, message):
    send_mail(
        subject,
        message,
        settings.DEFAULT_FROM_EMAIL,
        [to_email],
        fail_silently=False,
    )

# Example Usage
send_email("donee@example.com", "New Food Donation Available", '
```

### Functionality:

- When a restaurant posts a donation, an email is sent to all selected donees.
  - Once an NGO accepts a donation, a confirmation email is sent to the restaurant.
-