Image Upload and Thumbnail Generation

Task.py

```
@shared_task
def send_welcome_email_async(id):
   time.sleep(5)
   photo = Photo.objects.get(id=id)
   img = Image.open(photo.image.path)
   thumbnail_size = (150, 150)
   img.thumbnail(thumbnail size)
   thumb io = BytesIO()
   img.save(thumb_io, img.format)
    thumb_io.seek(0)
    thumbnail_name = f"thumbnail_{photo.id}.jpg"
    photo.thumbnail.save(
            f"thumbnail/{thumbnail_name}",
           ContentFile(thumb_io.getvalue()),
            save=True
    send_mail(
        "Welcome!",
        "Thanks for uploading a photo! This mail was send asynchronously.",
        "from@example.com",
        ["to@example.com"],
        fail_silently=False,
```

Models.py

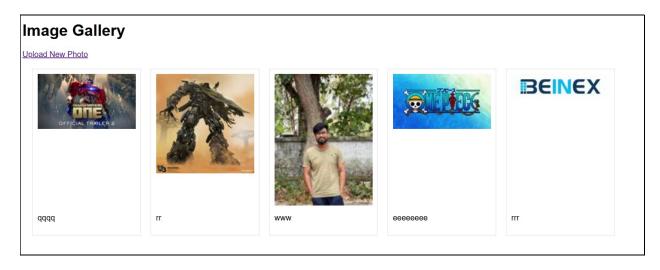
```
class Photo(models.Model):
    title = models.CharField(max_length=100)
    image = models.ImageField(upload_to="photos/")
    thumbnail = models.ImageField(upload_to="thumbnail/",null=True,blank=True)

def __str__(self):
    return str(self.title)
```

views.py

```
def upload_photo(request):
    if request.method == "POST":
        form = PhotoForm(request.POST, request.FILES)
        if form.is_valid():
            photo = form.save()
            send_welcome_email_async.delay(photo.id)
            return redirect("gallery")
    else:
        form = PhotoForm()
    return render(request, "gallery/upload.html", {"form": form})
```

Landing page



Upload file page



Mailtrap page

