Task - 2.5

Changes made to Model Database:

```
class Recipe (models.Model):
  name = models.CharField(max_length=255, blank=False, help_text="Enter the name of the
recipe")
  ingredients = models.TextField(help_text="Insert ingredients separated by a comma")
  cooking_time = models.PositiveIntegerField(validators=[MinValueValidator(1)], help_text="Insert
the cooking time in minutes")
  difficulty = models.CharField(max length=30, editable=False)
  method = models.TextField(help_text="Describe the cooking method of the recipe")
  pic = models.lmageField(upload_to='recipes', default='no_picture.jpg')
  liked_by = models.ManyToManyField(User, related_name='liked_recipe', blank=True)
  def calculateDifficulty(self):
     items = [i.strip() for i in self.ingredients.split(",") if i.strip()]
     if self.cooking_time < 10 and len(items) < 4:
       self.difficulty = "Easy"
     elif self.cooking_time < 10 and len(items) >= 4:
       self.difficulty = "Medium"
     elif self.cooking time >= 10 and len(items) < 4:
       self.difficulty = "Intermediate"
     elif self.cooking_time >= 10 and len(items) >= 4:
       self.difficulty = "Hard"
  def save(self, *args, **kwargs):
     self.full_clean()
     self.calculateDifficulty()
     super().save(*args, **kwargs)
  def __str__(self):
     return str(self.name)
```

- Add 'pic' attribute
- Changed 'likes' with many to many relation model attribute 'liked_by'. This allows each user to set one like or not to a specific recipe

JOURNAL DOC

Frontend Inspiration

https://www.giallozafferano.it/

What I like mostly of these page is the easy and straightforward layout, displaying only the essential infos of the recipe.

https://www.simplyrecipes.com/lunch-recipes-5091263

In these page I like the grid displaying the recipes with the category search bar on top

