Модели и план приложения foodgram.

**Приложение users.**

Пользователи – Users, отдельное приложение + использование стандартного инструментария django.

**Приложение recipes.**

Recipe. Поля, обязательны:

* author (автор рецепта) – **ссылка** на модель User, при удалении пользователя удалятся связанные с ним рецепты
* name (называние рецепта) – чарфилд
* image (изображение рецепта) – имеджфилд
* description (описание рецепта) – текстфилд
* image (изображение рецепта) – имеджфилд
* ingredients (ингредиента для рецепта) – **ссылка** на модель Indgredients, выбор ингредиента, количества, единицы измерения, при удалении ингредиента связанные с ним рецепты не удаляются
* tag (тег рецепта, завтрак обед ужин) – **ссылка** на модель Tag, при удалении тега связанные с ним рецепты не удаляются
* time (время приготовления рецепта в минутах) – таймфилд ? посмотреть подходящее поле

Ingredients. Поля, обязательны:

* name (называние ингредиента) – чарфилд
* quantity (количество ингредиента) – числовое поле, возможно с ограничениями
* unit (единица измерения ингредиента) – **ссылка** на модель Unit, при удалении единицы измерения не удаляются связанные с ней ингредиенты

Unit. Поля, обязательно:

* name (наименование единицы измерения) – чарфилд

Tag. Поля, обязательно и уникально:

* name (называние тега) – чарфилд
* color (цвет тега) –? колорфилд
* slug (слаг тега) – слагфилд

Follow. Поля:

* follower (кто подписан) – **ссылка** на модель User, форенкий
* following (на кого подписан) – **ссылка** на модель User, форенкий

Favorit. Поля:

* user (пользователь) - **ссылка** на модель User, форенкий,
* recipe (рецепт) - **ссылка** на модель Recipe, форенкий

**Приложение shopping\_list.**

List. Поля:

* user (пользователь) - **ссылка** на модель User, форенкий
* recipe (рецепт) - **ссылка** на модель Recipe, форенкий

**Приложение api.**

Весь api в отдельном приложении.

UNIT\_CHOICES = (

'банка',

'батон',

'бутылка',

'веточка',

'г',

'горсть',

'долька',

'звездочка',

'зубчик',

'капля',

'кг',

'кусок',

'л',

'лист',

'мл',

'пакет',

'пакетик',

'пачка',

'пласт',

'по вкусу',

'пучок',

'ст.л.',

'стакан',

'стебель',

'стручок',

'тушка',

'упаковка',

'ч.л.',

'шт.',

'щепотка'

)

**import csv**

**from django.core.management.base import BaseCommand**

**from recipes.models import Ingredient**

**class Command(BaseCommand):**

**help = ('Добавляет данные из csv файла директории /data/ '**

**'в базу данных sqlite3. Перед добавлением удаляет все записи '**

**'используемой модели!')**

**def handle(self, \*args, \*\*kwargs):**

**with open('/Users/eugenia/Dev/foodgram-project-react/data/ingredients.csv', 'r') as csv\_file:**

**rows = csv.reader(csv\_file, delimiter=',')**

**header = ('name', 'measurement\_unit')**

**Ingredient.objects.all().delete()**

**for row in rows:**

**object\_dict = {key: value for key, value in zip(header, row)}**

**Ingredient.objects.get\_or\_create(\*\*object\_dict)**

{

"ingredients": [

{

"ingredient": {"name": "Картофель"},

"quantity" : 2.0

},

{

"ingredient": {"name": "Слоеное Тесто"},

"quantity" : 700

}

],

"tags": [1],

"image": "data:image/jpeg;base64,",

"name": "Рулет Венлингтон5",

"text": "Британская рождественская классика",

"cooking\_time": 100

}