Initial Recipe Model (Exercise 2.3)

- **Model: Recipe**
- **Attributes:**
    - **recipe_id**: Primary Key (Auto Incremented)
    - **name**: Character Field (Max Length: 300)
    - **description**: Text Field
    - **cooking_time**: Integer Field
    - **ingredients**: Text Field
    -

Updated Recipe Model

- **Model: Recipe**
- **Attributes:**
    - **name**: Character Field (Max Length: 300)
    - **description**: Text Field (Max Length: 500)
    - **cooking_time**: Float Field (with help text "in minutes")
    - **ingredients**: ManyToMany Relation with **Ingredient** model
    - **image**: Image Field (with upload path "recipes/")
    - **creation_date**: DateTime Field (auto-added on creation)
- **New Model: Ingredient**
- **Attributes of Ingredient:**
    - **name**: Character Field (Max Length: 100, Unique)

Reasons for Changes:

- **Removal of recipe_id**: Django automatically adds an **id** field as a primary key. Explicit declaration was redundant.
- **Addition of max_length in description**: To maintain data consistency and limit the size of the description.
- **Change in cooking_time data type**: Switched from Integer to Float to allow for more precise cooking times (e.g., 5.5 minutes).
- **Introduction of Ingredient model**: To normalize the database. Ingredients are now stored separately and can be associated with multiple recipes.

- **Addition of image field**: To allow users to upload a visual representation of their recipes.
- **Addition of creation_date**: To track when a recipe was added to the database.
- **Method calculate_difficulty**: To dynamically assess the difficulty level of a recipe based on its cooking time and the number of ingredients.

These changes were made to enhance the application's functionality, improve data normalization, and provide a richer user experience by allowing more detailed and structured data to be stored.