Mapping of relations to SQLalchemy datatypes

SQLalchemy datatypes are in parentheses at the end

**Domains and Constraints:**

**Categories:**

* CategoryType (Primry key, not null) (Int)
* *recipeID* (foreign key, primary key, not null) (BigInt)

**Recipes:**

* Recipe id: not null, unique – put bigInt in assumption that the recipe website will have a large number of recipes (BigInt)
* Cook time: integer? Float?, not null (integer in minutes?) (Float)
* Name: varchar, not null (String)
* Description: varchar, not null (Text)
* Servings: integer(?), not null (Integer)
* Difficulty:( “easy”, “easy-medium”, “medium”, “medium-hard”, “hard”) default value: “medium”, not null (ENUM)
* Steps: integer – how many steps a recipe has, not null (Integer)

**Ingredients:**

* IngredientID: not null, unique (PRIMARY KEY) (Integer)
* Name: (varchar) not null (String)
* Aisle: varchar, not null (String)

**User:**

* UserID: not null, unique PRIMARY KEY (BigInt)
* Username: not null, unique (String)
* Creation Date: date format, not null (Time)
* Password: not null, unique (String)
* Last Access Date: date format, not null (Time)

**userPantry:**

* + PantryItemId – primary key, unique, not null (Int)
  + ingredientId – foreign key, not null (Integer)
  + userID – foreign key, not null (BigInt)
* expirationDate: date format (Time)
* Purchase date: date format, not null (Time)
* Current quantity: (float in terms of weight?) not null, Default = 0 (Float)
* QuantityBought: (float in terms of weight?) not null, Default = 0 (Float)

**Recipe\_Ingredients:**

* Amount: (float in terms of weight?), not null (Float)
* *recipeID* primary key, foreign key, not null (BigInt)
* *IngredientID* (primary key, foreign key, not null) (Integer)

**cookedBy:**

* *recipeId* unique, not null, foreign key, primary key (BigInt)
* *userId* unique, not null, foreign key, primary key (BigInt)
* Rating (not null, integer) (Integer)
* Scale (default = 1) (Float)
* cookDate date format, not null (Time)

**Steps:**

* *recipeID*  (unique, primary key, foreign key, not null) (BigInteger)
* Stepnr (not null, primary key) (Integer)
* Instructions (not null) (Text)