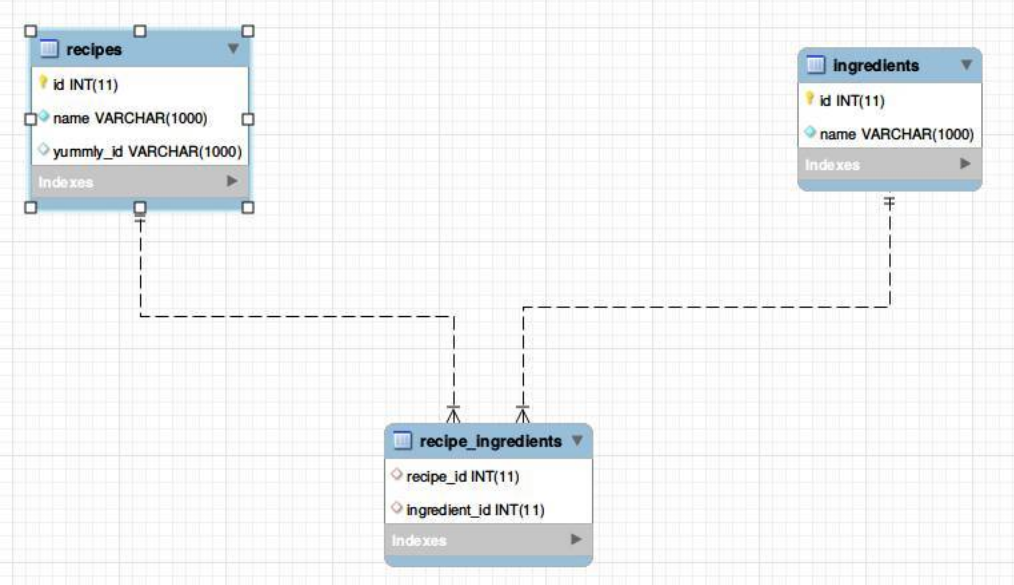
Sofia Mehrotra
Micheal Barlow
Erin Ruby
Rasheeq Jahan
Greg Benton

We have selected MySQL and MySQL Workbench as our software, and have used python scripts along with Yummly to construct our database. Yummly is a recipe aggregation website with an API that allows us to search recipes based on attributes and get recipes based on ID's. By calling the API a few hundred times we have been able to pull data from the website and fit it into a database.

The database is composed of three main tables: recipes, ingredients, and recipe_ingredients. The first table holds recipes with id, yummly_id, and name fields where yummly_id is the id that can uniquely be used to retrieve the recipe data from the API. The ingredients table has fields name and id, and the recipe_ingredients table has recipe_id and ingredient_id as fields.

The way this is set up is that any row in the recipe_ingredients table will correspond to one of the ingredients that is contained in the recipe with the associated id. This will allow us to look us both ingredients based on recipes and recipes based on ingredients which are two of the functionalities that are necessary for our website.

- The database design diagram is included below.
- The SQL script that is used to generate the database (reverse engineered using MySQL Workbench) is located here: `https://github.com/g-benton/3308-Project/blob/master/recipe_db_hard_backups/recipe-db-backup.sql`
- The python scripts used to populate the database with data from the Yummly API can be found here: `https://github.com/g-benton/3308-Project/tree/master/populate-db-yummly`



EER Diagram of the database