**Design and data storage for Project: Recipe Holder**

You will need to have three separate Dialogue windows: one for adding each Ingredient to the Ingredient list (including number, name, and amount), and one each for adding the Instructions as a step-by-step method or as one large text block (at the user’s choosing).

INGREDIENTS

For the ingredients, a new Dialogue should pop up every time a new Ingredient is added. For the Ingredients, create a separate Table in recipe.db with columns for Number, Ingredient Name, and Amount. You will just have the app save each of these to recipe.db and retrieve each of these items directly from this table in recipe.db when the app is re-opened. In order to format this text window, you will probably want to use an HBox and add smaller text areas for each of these fields. You may also need to find a way to lengthen these fields so that everything fits on one line, if possible, which means you’ll have to rearrange these text areas in relation to the ListView that stores the Recipe names.

TableName: ingredients

Columns: ingredient\_ID, name, amount, instructions\_ID

INSTRUCTIONS

For the Instructions, the easiest method would be to allow the user the option of entering the instructions in a step-by-step format (composed of step number (actually, just have the app set the number automatically) and the text for the instruction (one for each step) or as a single text description). If the user chooses a step-by-step, store the number and text instruction in a STEP-BY-STEP table in a recipe.db with columns NUMBER and INSTRUCTION and just save the info to and retrieve it from recipe.db.

If the user chooses to enter as one large text block, store and retrieve the text block as it was entered by the user as a text file in the same local directory where the app files are located (you’ll need to check there’s enough memory available). Don’t worry about formatting any of this text, but do allow an option for the user to redo and re-save this text. You will then save the “directory + filename” for each recipe item within another Table in recipe.db. The app will use this information to retrieve the text blocks from the directory location.

TableName: instructions

Note: Initially, the step\_number will only be 1. This will change once we add an option for user to input the instructions step-by-step rather than one chunk of text (which will be saved to a file stored in the user’s local directory).

Columns: instructions\_ID, step\_number, ingredient\_ID