Project Part 3

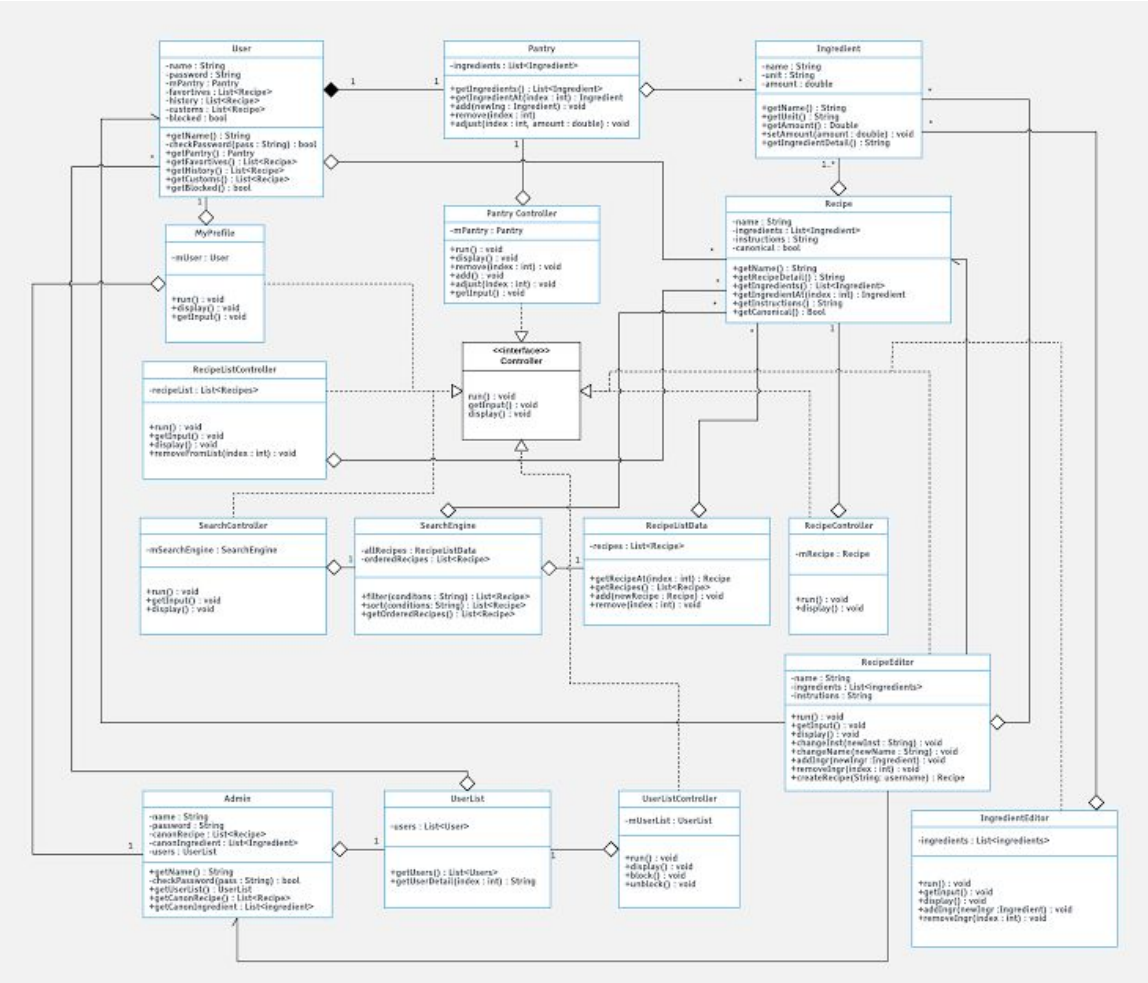
Team: Xiaolan Cai
Chi Chen
Andrew Guttman

Title: Ingredient Tracker

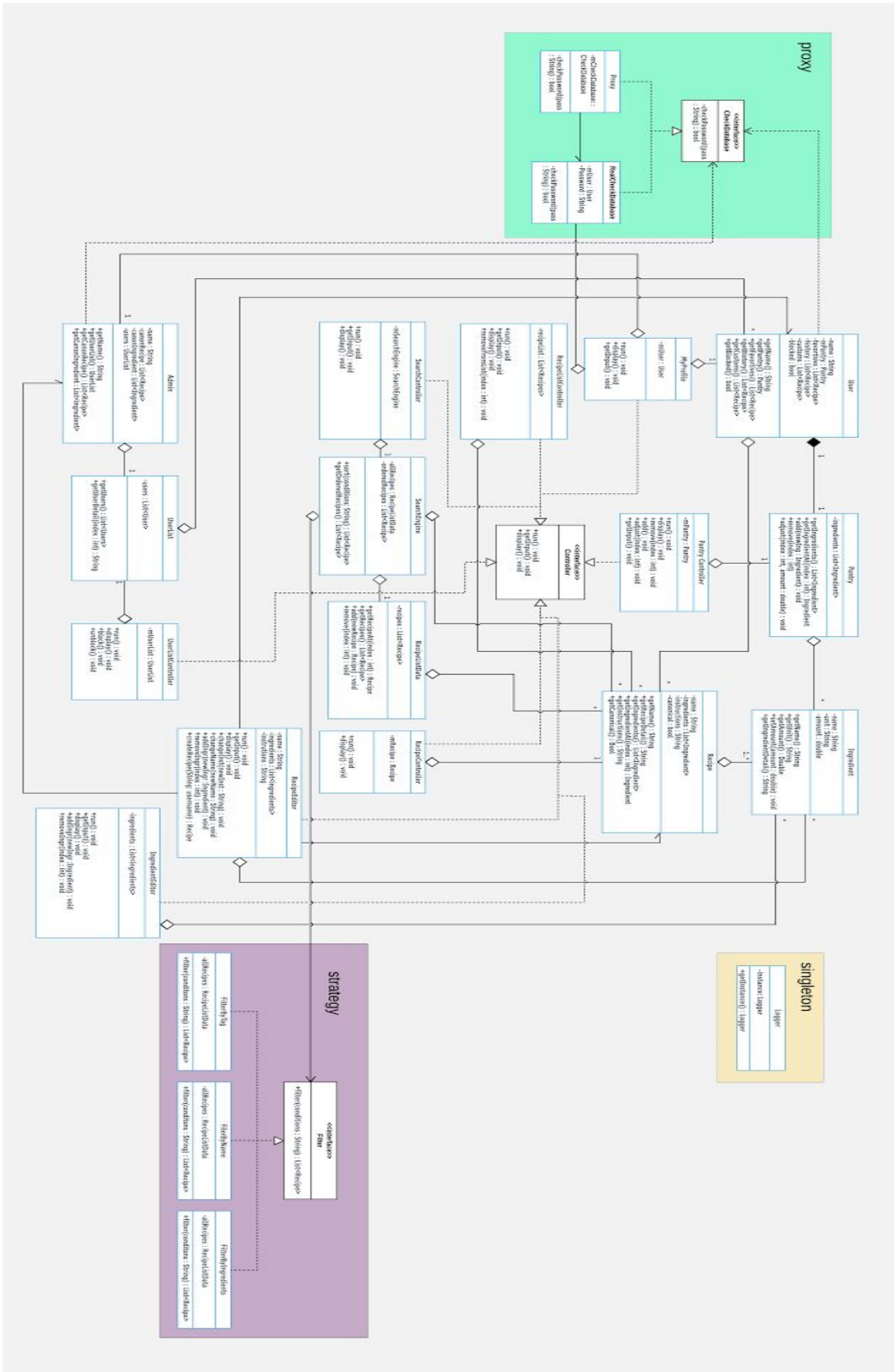
Description:

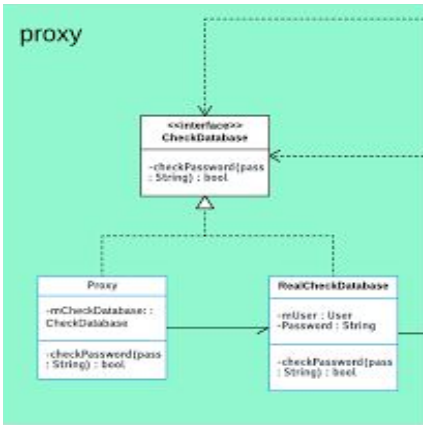
A web application that allows users to keep a digital pantry of ingredients, create and share recipes and search/filter/sort through recipes based on available ingredients and user metrics.

Class Diagram Before:

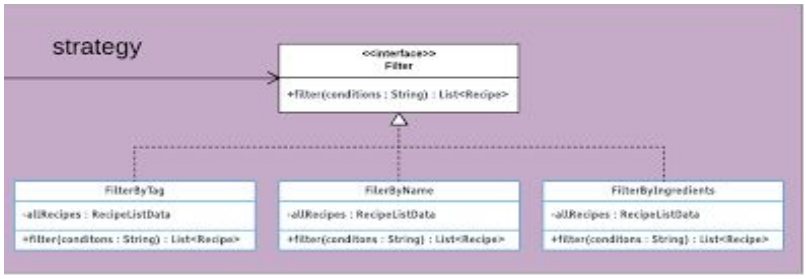


Class Diagram After:

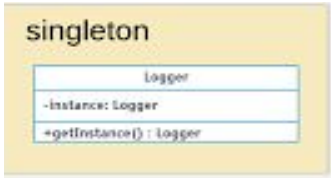




First, as you can see, we apply the Proxy patterns, we removed the checkPassword() from User and Admin, and implement this function with a Proxy pattern. We mark the part we change into green in this part. Using an interface, “Proxy” class and “RealCheckDatabese” class to implement the Proxy pattern. The reason we think it might be better is that when checking the password, we need to access to the database, and it is exactly when we should use Proxy pattern to protect our system.



Another part we change is that we apply the Strategy patterns, we removed the filter() from SearchEngine, and then implement this function with a Strategy pattern. We mark the part we change into purple in this part. Using a interface “Filter” and three classes for three ways implementing filter to realize the Strategy pattern. The reason we think it might be better is that we have three ways to filter the recipe, and Strategy pattern can help us here to classify three way for us.



In the end, we think we should have a logger to log all the action in our program, and if we have more than one user, the logs might not be as clear as we want, but we want that one log should map to one and only one action. For this function, Singleton pattern will be the best pattern to apply. We mark the part we change into yellow in this part. Using a class name “Logger” and a method in this class to implement Singleton pattern.