Problem Domain: Cooking for larger groups of people. Cooking for groups larger than the size of a typical recipe can be challenging for various reasons. One of the main issues I face is deciding how many ingredients to prepare. This problem is specific to me because I live in a dorm where I cook for about 30 people every other week. This is usually a fun experience, but preparation can be a bit difficult since most recipes are made for at most eight people. In addition, I have learned to cook recipes in different ways when I'm cooking for larger groups. This domain covers me cooking for my dorm, but could also apply to large events with food, like parties.

Problem: Scaling recipes. Scaling up the ingredient usage is not as straightforward as it might seem due to some ingredients needing different amounts when scaled up. For example, if I'm making a recipe for five times the number of people recommended, not all ingredients should be scaled by five due to the way that most things can be reused.

Stakeholder List:

- Event Chefs: These are cooks who cook for large groups of people and need to scale their recipes. These stakeholders are potential users of this software who can be used to gauge the effectiveness of the software.
- Community Cooks: People like me who cook for communities with large numbers of people could also be users of this app.
- Family Events: Families who cook for large gatherings, such as Thanksgiving, could also take advantage of this software.

Evidence and comparables:

- www.webstaurantstore.com This link is an example of a current recipe converter that does nothing but multiply each ingredient amount by the scale given. This could be useful, but my software would be more specific to each ingredient and intelligently scale each ingredient.
- Facebook forums. There are various forums discussing how certain ingredients like spices, do not need to be scaled as much as others due to how dishes are cooked.
- www.cooksdelight.com/blog/4-easy-steps-to-scale-up-a-recipe-for-foodservice-menus/ This link emphasizes that scaling recipes is not an exact science and every ingredient needs its own scaling factor.
- ibakemistakes.substack.com This blog has some threads that speak on how baking also has this problem with scaling recipes.
- www.quora.com/What-are-your-tips-for-scaling-recipes-up-or-down This thread speaks on some specific ingredients that have certain ways to be scaled up or down.

Application Pitch: Scale & Savor

This application will solve the problems associated with scaling recipes up for larger groups or down for single cooks.

- The main feature will be the **scaler** that will allow users to manually enter or upload recipes that can then be scaled by the factor that they choose. This will be the main purpose of the software, which with enough precision, should mitigate the problem of difficulties scaling recipes for larger groups or single-person dishes. This feature should be the main one used by the stakeholders and, therefore, should be very easy to use and understand.
- Personalized **tips for scaled cooking** will be another feature that goes along with the scaler to provide tips for how to cook smaller or larger amounts, depending on the input recipe. This feature will be less exact than the scaler, but will provide optional tips on how to change recipe steps to fit your scale factor. This should impact most stakeholders as a large change in the scale of recipes will probably result in some change in the procedure for cooking.

Concept Design:

concept Scaler

purpose meant to store recipe and scale to the input factor

principle recipe either manually inputted or uploaded along with scale factor
on scale, ingredient counts are scaled by the given factor

state

a set of recipes
a name
a scale factor
a set of ingredients
a set of cookingMethods

actions

enterRecipe (name: String, originalPeople: Number, targetPeople: number, ingredients: List[Ingredient], cookingMethods: List[String]): ()

requires targetPeople to be a whole number between [1, 50] and name to be unique

effect creates a scale factor based on the original number of people and the target amount and enters this recipe into the set of recipes. Optional cooking methods can be added to be associated with the recipe.

scaleRecipe (name: String): (result: List[Ingredient])

requires name to be in set of recipes

effect returns the list of ingredients after being scaled

concept Ingredient
purpose stores information on a certain ingredient and how to scale it
principle created along with a recipe
when a recipe is scaled each ingredient is independently scaled
state
a set of Ingredients
a name
context on scaling
actions
scaleIngredient (scaleFactor: Number, originalAmount: Number): (newAmount: Number)
effect uses context to intelligently scale the amount of the ingredient given

concept tipsPage
purpose stores an increasing number of tips for certain aspects of cooking
principle dynamically changing board of tips for scaled cooking
state
a set of Cooking Methods
a scaling up tip (optional)
a scaling down tip (optional)
actions
addTip (cookingMethod: string, up/down: Boolean, tip: String): ()
effect updates the tip for scaling up or down for a specific cooking method
requestTip (cookingMethod: string, up/down: Boolean): (tip: String)
effect returns the tip associated with that cooking method for scaling up or down

sync ingredient scaler
when Scaler.scaleRecipe(recipe)
then Ingredient.scaleIngredient(ingredient for ingredient in Scaler.recipe.ingredients)

sync cookingMethodSearch
when Scaler.scaleRecipe(recipe)
then tipsPage.requestTip(method for method in Scaler.recipe.cookingMethods)

The concepts control the main flow of the software application. The main concept is the Scaler, and all other concepts that would come after will be built off the Scaler. The state is very dependent on each recipe and is easily scalable. For example, the ingredients will vary from recipe to recipe, but each general ingredient will have some context for scaling that will be stored in a separate Ingredient context. This is meant to separate the ingredients database and the recipes database for each user.

The cooking methods part of the Scaler concept is optional, but will help the user throughout the cooking part of their recipe, in addition to the ingredients. In practice, when a user uploads a recipe, that recipe can be scanned for cooking methods that can be automatically generated when scaling. This is associated with a separate concept that, similarly to the ingredient concept, can be added to and changed to store information for each cooking method and how that can change when scaling.

UI Sketches:
Reference image linked in README.md

User Journey:
Imagine that Victor decides to hold a party for his large extended family. He wants to make sure that everyone is fed at this party, but he also wants to make sure that he buys exactly what he needs, as he is on a budget and doesn't want leftover ingredients since he doesn't cook very often. He finds a recipe online for chicken tacos, but it says it feeds about six people when he expects 30 people to eat at this party. This is when he discovers Scale & Savor!

He decides to enter the recipe he finds into the application. He is able to download the recipe he finds as a PDF and easily upload it to the app. The software then reads the PDF and automatically fills in all the necessary information to get started scaling his recipe. He also sets how he plans to cook the dish so the app can give him tips on how to cook large amounts of food in his regular kitchen. With this done, he sets the target amount to 30 people, and within a few seconds, his recipe is scaled up!

Victor is then able to confidently go grocery shopping and stay under his budget. With the tips that the app provides, he makes the most of his small kitchen and has a successful party for his entire guest list with no leftovers! After this experience, Victor is excited to keep using Scale & Savor every time he hosts large events.