

Group Name: **Chef Devs**

Adeeb Khan, Samuel Li, and Daniel Kibatullin Period 9

Project Manager: Adeeb Khan

Developers: Samuel Li and Daniel Kibatullin

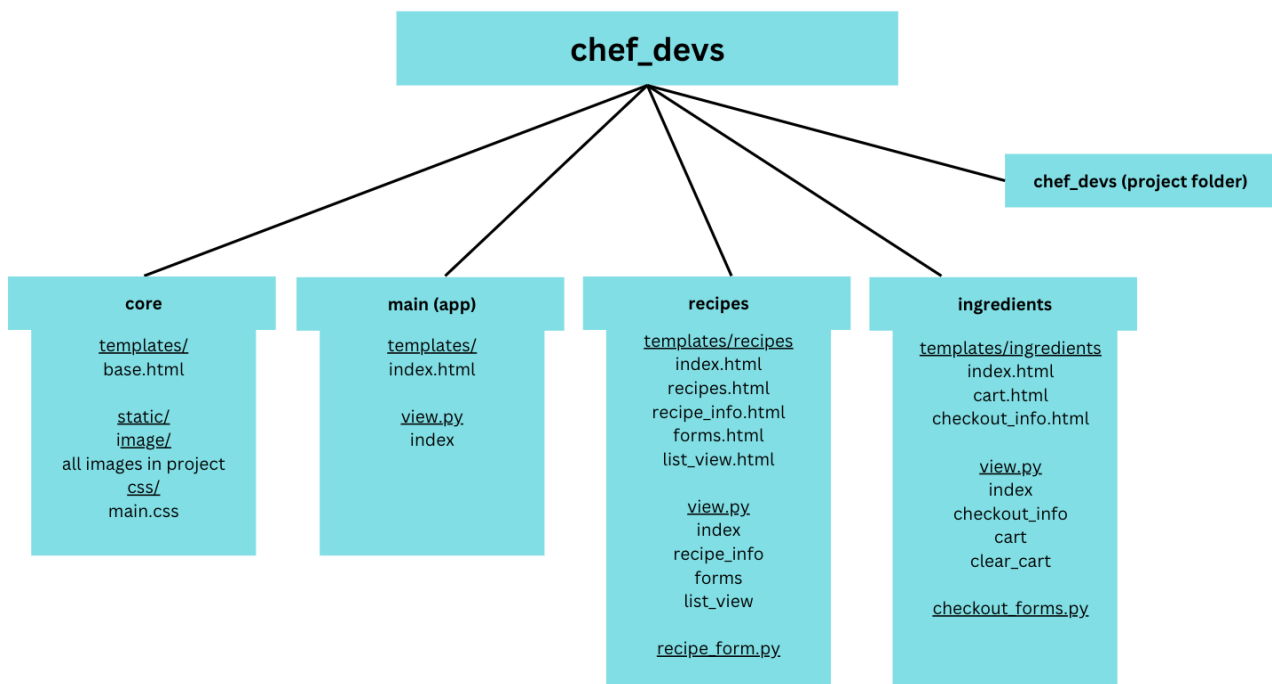
Description

Chef Devs is a website where a user will be able to access a recipes app and an ingredients app. In the recipes app, the user will have access to a list of foods that they can then search the recipe for. Upon searching the name of a food from the list, the user will be directed to a page where they will be able to read the recipe on how to make that given food. If a user wants to add their own recipe, they may click the “Update the Recipes!” link which will lead them a page with 2 forms. One form allows for the creation of a new recipe by typing in the food name and recipe while the other allows you to exist any recipe from the list of recipes. Beneath the “Update the Recipes” link is a link that shows the original form working the same way. The only exception is that this original form does not use django forms and is generated from raw html.

In the ingredients app, the user is able to view a table of available ingredients that has the current stock of the ingredients, price, and name. In the form on this app, the user can type in the name of the food that they want to buy. They will then be directed to an ingredient_info page where they can specify the quantity and add it to the cart. Upon making their purchase, it can be seen that the stock quantities available change relative to how much stock of that ingredient you bought.

To tie it all together there exists a main app that serves as the home page for our project. This project allows the user to be directed to either the recipes or ingredients app while also having a link to our git repository.

General Structure (key files only)



Development History:

Oct 6: Project is initialized, all apps and files being created.

Oct 7-10: Adding our primary data structure to the views (large dictionary of foods and recipes)

Began updating url pathing between sites, created templates, fixed pyenv and merge bugs.

MAJOR DECISION: Adeeb lays out general structure of the project and tasks Daniel and Sam to work on data structures while he initialized files and apps.

October 11: Began creating forms and cookies for recipes. Major bug fixes. Added additional templates for the form to process input data. Major updates to templates html files allowing it to have access to our dictionary and list all food names while also showing form. Form is not functional yet however. Adding static files such as images and css files for base.html and adding corresponding images for EVERY food in the recipes list dictionary. Initializing all templates for ingredients as well.

MAJOR DECISION: previous idea was to have a comments app but this turned out to be problematic. After holding a meeting, Adeeb decided to add an ingredients app instead and put Daniel and Sam in charge of it.

October 12: Fixed same site cookies for ingredients and recipes. Added functional forms that allow user to go to recipe_info and have contents of recipe_info be dynamically updated based on what food the user searched for. Images are able to be extracted from core into recipe_info if a user requested recipe for a given food. All image urls have been added to every food in the dictionary. Website logo created and displayed in base.html. Functional forms for ingredients added where stock available in main ingredients table is able to be dynamically updated based on user input. JSON implementation failed leading to major bugs. **MAJOR DECISION:** Adeeb will finalize primary features in recipes, while Sam will finalize primary features of ingredients while Daniel will finalize list_view and additional forms views.

October 13: Stylistic features added to base.html adding color to the webpage while also adding a nav bar. Main app created that acts as official homepage for the site. JSON dumps added to primary forms in ingredients and recipes. Table in ingredients is styled properly using flex box. URL navigations between app urls are added through links on templates. Nav bar styles added so that it is applicable to every app other than main. Updatable forms are created. list_views allows user to add recipes to recipes list through a form and delete them. Non Django version of primary recipes form is created and displayed using link. All url pathing is finalized and every app can connect to each other. Bug fixes in ingredients and recipes, finalizing messages displayed on apps. Adding larger text label to update form in list_views to allow user to update dictionary with their own recipe. Finalization of styles and forms. Website is fully functional and meets all requirements posted.

Work Distribution 10/10 🍌

Adeeb: In charge of creating the entirety of recipes app and it's related views. Created all relevant templates and forms for recipes. In charge of deciding what structure of project should be. Modified all apps in projects to fit project structure. Added general stylistic features for all apps. Implemented JSON for all primary forms in ingredients and recipes. Created all cookies for recipes. Authored the working document and held all meetings to direct developers on what they should be working on.

Sam: In charge of creating entirety of ingredients app and its related views. Created all relevant templates and forms for ingredients. Added flex box features to ingredients for better view quality. Worked with Daniel to create cookies for ingredients. Created cart feature in ingredients allowing user to view cart items.

Daniel: Created data structures for ingredients and recipes and individually got images for every recipe. Made general outline of ingredients app. Made entire list_views for recipes creating updatable forms that allows user to update recipes dictionary. Created non Django version of recipes form. In charge of specific stylistic features for apps all around.

Overall every member did their part and there were no conflicts. Every member was able to finish their part of the project in an orderly and timely manner and were able to communicate with each other effectively to get work done and minimize merge conflicts. Everyone contributed to fixing bugs and all parts of the project are complete. No one person did too little or too much work. Members were able to work together after school at a designated location and were also able to communicate virtually to get work done.

