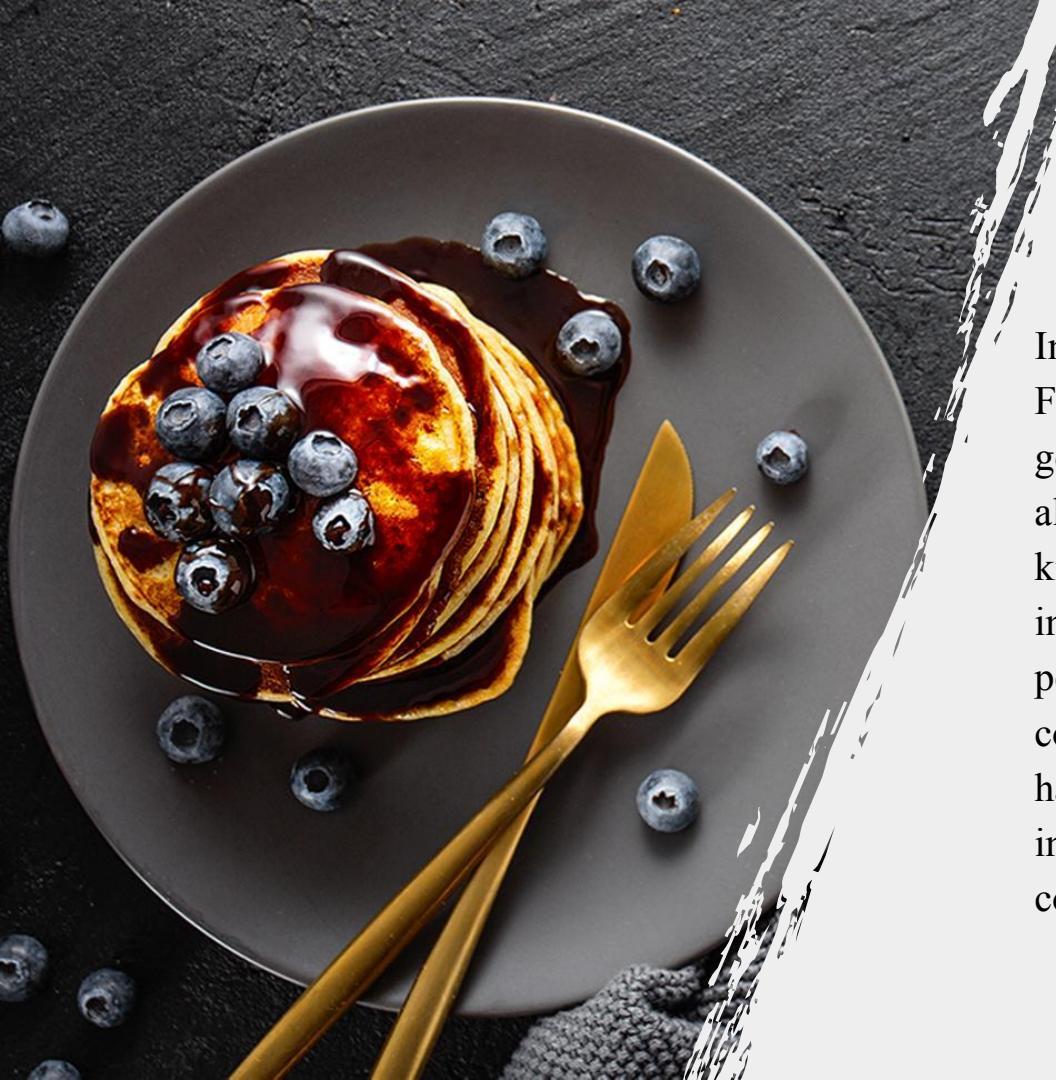




COOKBOOK

By
Aaron Amha,
Alex Foucher,
Michael Dempsey.
Sarah Sharroufna

A top-down photograph of a stack of golden-brown pancakes on a dark grey plate. The pancakes are topped with a generous drizzle of dark syrup and fresh blueberries. A shiny gold fork lies next to the stack. The plate rests on a dark, textured surface. Scattered blueberries are visible around the plate.

Description Of The Project

Inspired by the most successful social media ever, Facebook, our group sought to provide foodies and general culinary enthusiasts a network that allows all of the limitless and creative ideas had in the kitchen, to come together on one platform. On our interface, registered users have their own personalized profiles which consist of their contributions to the network, as well as recipes that have captivated their interests. Users are allowed to interact with other recipes on the platform with comments which fosters a sense of community.



FULL COURSE FEATURES

APPETIZER: REGISTER / LOGIN PAGE

Serves as a check to determine whether the user has created an account or not. If they have, they are redirected to our 'discover' page. If not, they are directed to the registration page, where they can create a personal account with a securely hashed password for protection.

MAIN DISH: DISCOVER / ADD RECIPE PAGE

The discover page showcases recipes from the Spoonacular API and recipes created by the user. Here, users can also favorite recipes, which are then stored on their 'favorites' page for easy access. Users can also comment their personal reviews on existing posts formulating an interactive environment. Interactive, as far as being able to view not only the user's profile, but even the person who commented's profile.

DESSERTS: LOG OUT

Which allows users to securely log out of their accounts on the website.



THE TOOLS USED BY OUR GROUP:

PROJECT TRACKING: OUTLOOK



Outlook constantly reminding us throughout the week about Varsha's weekly project meetings, as well as team updates to the github repository, gives it...

Ranking:

VERSION CONTROL (VCS) REPOSITORY: GITHUB SYSTEM



Despite its prominence in this industry, our project's Github Repository presented a plethora of merge errors. Code pulled for some members would be fully functional while others had issues.

Ranking:



THE TOOLS USED BY OUR GROUP:

DATABASE: POSTGRESQL



Granted our Database Configuration and Connection, as well as pg-promise libraries and create.sql all either say .env.POSTGRES or use PostgreSQL syntax, we can say without a doubt that our database was POSTGRESQL. Issues with this Database were mainly its complexity and lack of efficiency in handling BLOBS or (Binary Large Objects). A BLOB, for those unfamiliar, is the conversion of picture/video attachments to binary for storage purposes. With an application that deals largely with picture uploads, this inefficiency is limiting... Otherwise, not too bad...

Rating:



Two photographs of modern kitchen islands. The left photo shows a white countertop with black bar stools. The right photo shows a dark wood-paneled island with built-in ovens and a large window above.

THE TOOLS USED BY OUR GROUP:

INTEGRATED
DEVELOPMENT
ENVIRONMENT

(IDE): VS CODE



Visual Studio Code

Quick, reliable, free...
Good ole VS Code

Ranking:





THE TOOLS USED BY OUR GROUP:

UI TOOLS (WIREFRAMES): FIGMA



Although it was only used during the initial development/planning phase to get an idea of the necessary pages and features, it set a major precedent for formulating of the UI. The initial plans made on FIGMA certainly had a major influence on the outcome.

Ranking:

★★★★★

THE TOOLS USED BY OUR GROUP:

DEPLOYMENT ENVIRONMENTS:

CLOUD DEPLOYMENT: AZURE



A phenomenal source of IaaS, PaaS, and SaaS cloud services, Microsoft Azure is a solid cloud services environment for deployment reasons. Downside, wayy too many settings to figure out during initialization without Lab #13.

Rating: A five-star rating icon consisting of five yellow stars of decreasing size from left to right.

CONTAINER DEPLOYMENT: DOCKER



No issues at all, a vital part in development as it provides access to easily accessible isolated environments. Efficient containerization.

Rating: A five-star rating icon consisting of five yellow stars of decreasing size from left to right.



THE TOOLS USED BY OUR GROUP:

APPLICATION SERVER: NODEJS



Node.JS is highly optimal in handling an immense amount of libraries improving the software development process. Frequent async/await usage!

Ranking:



EXTERNAL APIs: SPOONACULAR



This API was close to perfect for our project as it not only provided a sufficient amount of recipes, but it even provided extra functionalities such as recipe searching and summarizing. The only recordable issue would be the API Rate Limit :(.

Ranking:





THE TOOLS USED BY OUR GROUP:

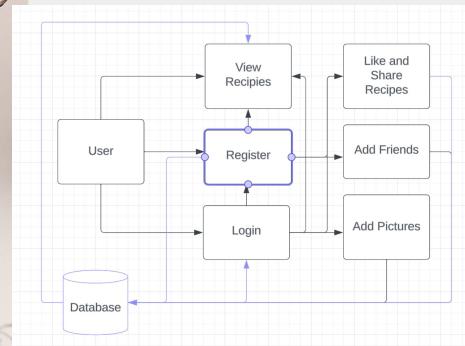
TESTING TOOLS: CHAI HTTP - NODE.JS



Chai HTTP provides an interface for live integration testing of the API's. The syntax behind it is easy to understand and implement, however, this process adds more seconds on Docker-Compose database/container initialization

Ranking:

FRAMEWORK: USER - DIAGRAM



We created our user diagram early on in the initial development phase of the project.

Two side-by-side photographs in a collage style. On the left, a woman with blonde hair tied back is stirring spaghetti in a large pot on a stove. On the right, a man with short brown hair is smiling while chopping a cucumber on a wooden cutting board. Both images have a torn paper effect at the edges.

METHODOLOGY

AGILE SCRUM FRAMEWORK

Prioritized list of features

Based on duration of task, importance to project, reliance of other tasks

Team members were assigned different tasks

Scrum master

Helped us maintain consistent level of work

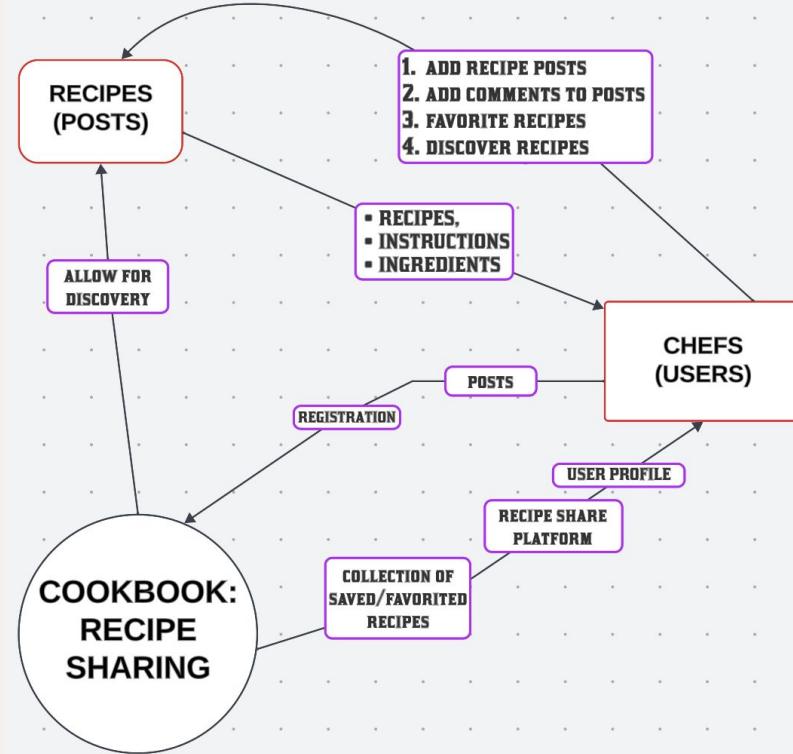
Ranking: 

SYSTEM ARCHITECTURE DIAGRAMS

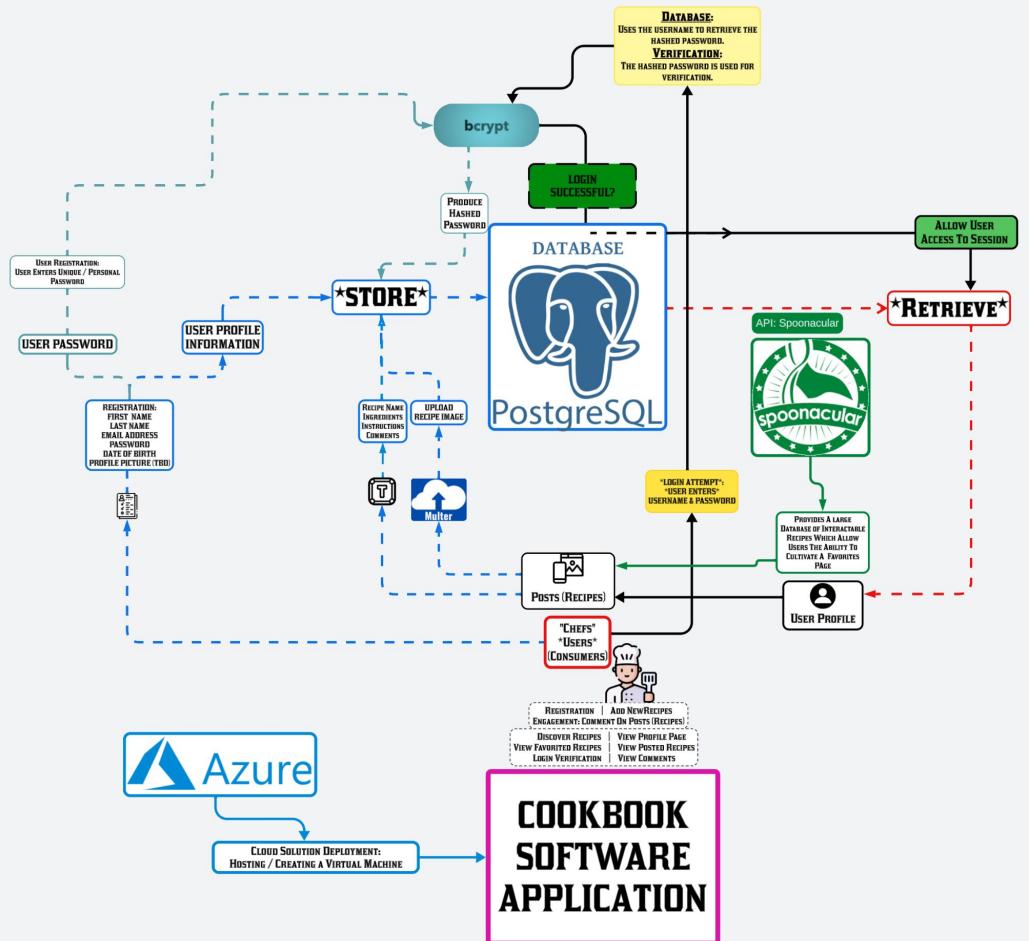




CONTEXT DIAGRAM



HIGH SOLUTION ARCHITECTURE DIAGRAM





CHALLENGES FACED

- One challenge we faced was merge conflicts. Everyone had their own work in different branches and more times than not, there were lots of errors that arose.
 - We resolved most merge conflicts by merging our branches together prior to pushing to main
 - We used the Merge Editor in VS Code to look at all the differences
 - Merge conflicts caused certain tasks to take longer than expected, which lowered the scope of the project
- Another challenge we faced involved tasks being dependent on other tasks.
 - For example, in order to have a profile page, the user database would need to be completed.



CHALLENGES FACED (CONT)

- We faced the challenge of delegating tasks. At the beginning, people were working on the same things, which led to us replacing code.
 - This caused tasks to take longer than expected
 - We overcame this challenge by assigning tasks and explicitly stating what we were going to work on