

# SafeEats

---

Make ordering your next meal safe, easy, and accessible

COMP 523 Team B: Mohammed Alnasser  
Reshmasai Malleedi  
Madison McFadden  
Rebecca Rozansky



01

## Recap

A brief reminder of our application.

02

## System Architecture

Our final design decisions: languages, tools, servers used.

03

## Working Demo

Walkthrough of our final end of semester product.

04

## Lessons Learned

What would we do differently, or the same next time?



01

What is  
SafeEats?

# Problem v. Solution

---

## I. Problem

Approximately 32 million people in the United States are affected by food allergies, but popular food delivery applications lack the initiative to accommodate people with allergies. Without accessible options to filter or select food with certain allergens, dining has become more difficult and inaccessible for a large community.

## II. Our Solution

*SafeEats* hopes to fix this issue by providing streamlined and accessible dining to those with food allergies.

- Users will be able to see the safe items of menus of nearby restaurants.
- Users can input dietary restrictions and preferences, and *SafeEats* will automatically parse the menus of restaurants for foods that meet the user's requirements.

# Platform Description

---

## Platform

- Mobile Application
- iOS
- Android

## Languages

- JavaScript

## IDE

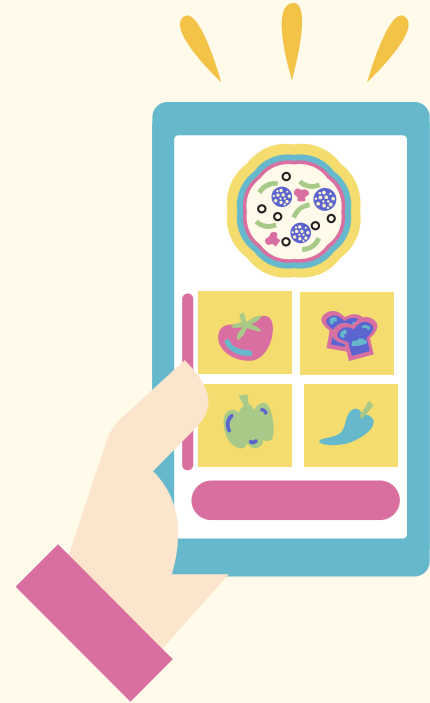
- VSCode
- iOS Simulator

## Code-Libraries

- React-Native

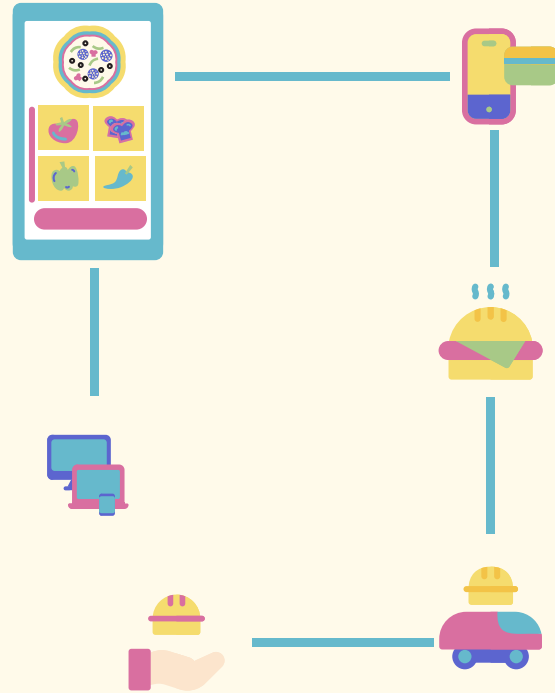
## Database

- Firebase



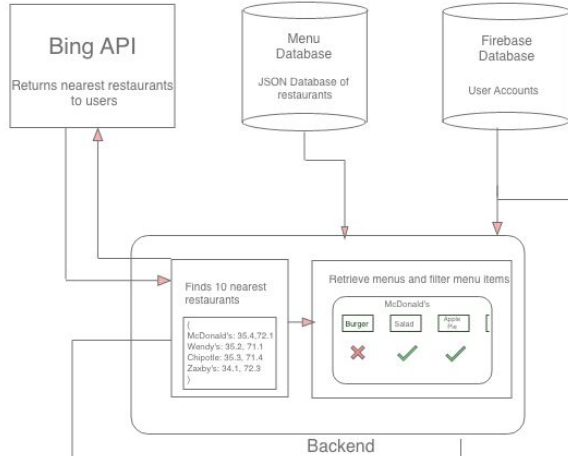
# 02

## System Architecture

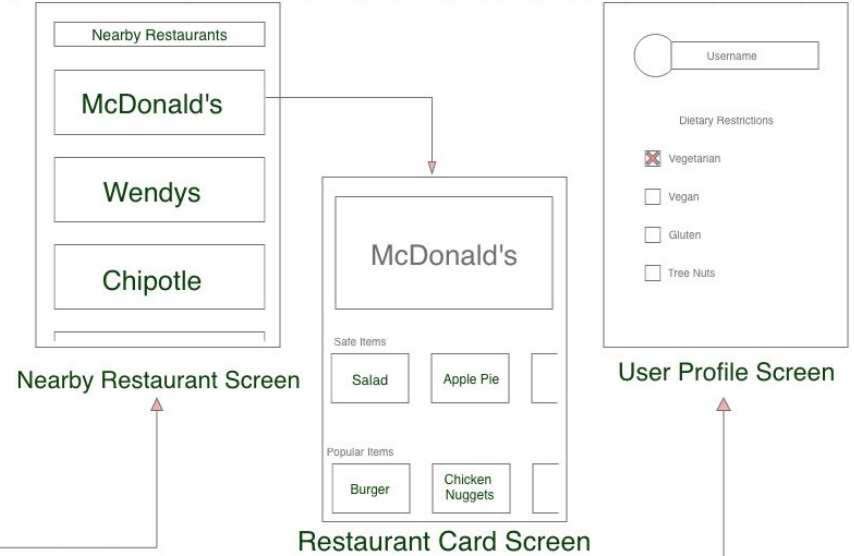


# Architecture Diagram

## BACKEND



## FRONTEND



# Technical Overview: Design Decisions

---

- **Nearby Restaurant API:** choosing between Google, Yelp, and Bing required considering factors such as features, cost, reliability, and ease of use. Google's API first came to mind but we quickly dismissed the idea because of the cost factor. Yelp's API, on the other hand, is free but allows a limited number of API calls per day. We settled on Bing's API because it provided us with everything we needed at no cost. Accessing the Bing API also had the least amount of security issues, Yelp required us to use CORS authentication.
- **Organization:** conflicts with our organization of restaurants and menu items - due to time limitations, we chose a faster way of storing this data in the form of a JSON file and manually inputting the data instead of automatically populating it. There was no consistent API that had restaurant menus that wasn't also free, therefore manually gathering menu data was our last option.
- **Cloud Storage:** we use Firebase realtime database to store the state of users in our database. This option was not the fastest, however, it's the cheapest when scaling across many users.



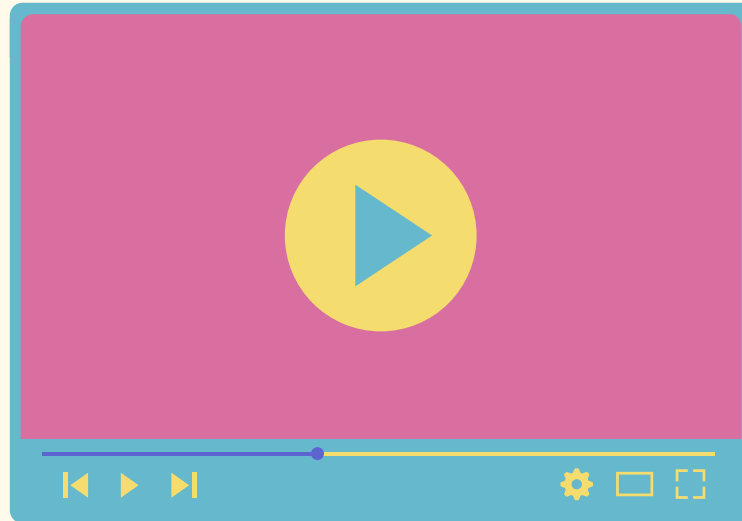


# 03

## Working Demo

# Demo

---



# 04

## Lessons Learned



# Key Takeaways

---

1. **Communication:** Effective communication is critical when working in a team. The presence of weekly meeting with our coach, client, and the team helped encourage us to stay productive and held us accountable in finishing our weekly deliverables.
2. **Project Management:** By meeting with our client, we were able to construct a plan for our application early on which aided us in the long run. We knew exactly what the application's purpose is, what features the client needed us to implement, and we knew our own skills so we did not take on more than what we were capable of.
3. **Collaboration and teamwork:** The team built a positive and supportive team environment, which helped us throughout this semester. We could, however, had improved our collaboration and teamwork by establishing clear roles and responsibilities and ensuring that each team member is accountable for their tasks.
4. **Prioritization:** Prioritizing tasks and features was critical to our progress. The team learned to prioritize tasks based on their impact and importance, allowing us to focus on the most critical components first.

# Things that went well & not as well

---

S

## Strengths

- Collaboration and teamwork among team members
- Access to resources like the TA and Client

V

## Weaknesses

- Conflicts in team members' experience and knowledge
- Broad scope

O

## Opportunities

- Gaining practical experience that can be used in future endeavors
- Developing a marketable and relevant application for the client
- Opportunities to learn new skills and technologies

T

## Threats

- Risks of burnout
- Client limitations to use certain technologies due to budget

# Questions?

