

Team 4 (Motor Munchies)

1. Olivia Lee

- **Student ID:** 030216509
- **Email:** Olivia.Lee01@student.csulb.edu

2. Matthew Marietta

- **Student ID:** 030178393
- **Email:** Matthew.Marietta01@student.csulb.edu

3. Luis Guillen

- **Student ID:** 030111833
- **Email:** luis.guillen01@student.csulb.edu

4. John Vo

- **Student ID:** 029774067
- **Email:** John.Vo03@student.csulb.edu

5. Ervin Delsas

- **Student ID:** 027020888
- **Email:** Ervin.Delsas@student.csulb.edu

6. Shawn Lin

- **Student ID:** 029766332
- **Email:** Shawn.lin01@student.csulb.edu

Preface

(Describe the purpose of this document, its expected readership and its version history, including a rationale for the creation of a new version and a summary of the changes made in each version.)

Version	Date	Changes
1.0	3/14/2024	Initial Version
1.1	15 th of April, 2024 (In Prog)	Added Glossary
2.0	43/15/24(In Prog)	Fixed version date notation, ...

Purpose

This document serves as a comprehensive guide for the development and understanding of the software project titled "Motor Munchies"

Audience

The intended audience of this document includes project stakeholders, developers, testers, and anyone involved in the project lifecycle.

Introduction

(Introduce the software project, its goals, and the problem it aims to solve)

Project Overview

"Motor Munchies" is a web-based food truck locator designed to assist users in searching for food trucks. It can be used to look up food truck vendors in a proximity, pinpoint a location as a vendor, and create reviews.

Project Goals

- ✧ Provide the location of various food trucks.
- ✧ Allow users to filter food trucks based on their preferences
- ✧ Allow users to post reviews and see other's reviews
- ✧ Display information of food trucks and their rating

Glossary

(Define key terms and acronyms used throughout the document, unless they are commonly known to each possible stakeholder (e.g., "Cell phone") AND used with their common meaning. Do not expect your stakeholders to be experts. If in doubt, define a term.)

- ✧ **GPS:** Global Positioning System
- ✧ **API:** Application Programming Interface.

User Requirements and Use Cases

(Outline what the system must do from the user's perspective. User stories need to use the format discussed in class and on our slides. Use cases provide detailed scenarios of system interactions.)

User Stories

1. As a traveling sales representative, I want a way to effectively look for local food trucks in my general location and select ones that pique my cravings so that I can plan around my busy work schedule.
2. As a food truck owner, I want to advertise my business so that customers are aware of the type of food I serve and where they need to go.
3. As a food blogger, I want a designated place to post my food reviews online whenever I eat at food trucks so that others can be informed about my experience.
4. As a sales representative with a very busy schedule, I want to be able to quickly filter the food trucks near me for a specific type of menu items or cuisine, so I can find specific food trucks around me to fit my dietary restrictions.
5. As a university student and gym newbie, I want to use Motor Munchies to find new, affordable, and healthy food so that I can stay within my budget and reach my fitness goals.
6. As a food truck enthusiast, I want to be able to pre-order foods from food trucks, so I can minimize wait times and pick up food quickly.
7. As a busy college student, I want to find food trucks that are quick to get to and be able to fit my budget to help me save money so I can get back to studying.
8. As a hungry customer, I wanted to find a good food truck so that I could order something to eat before I had to perform.
9. As a university event coordinator, I want to be able to find and book food trucks for campus events, so attendees have a variety of food options to choose from.

10. As an aspiring food critic, I want to be able to find food trucks around the LA area so that I can begin to write meaningful reviews that will get people to recognize me.

Use Case: Locating Nearby Food Trucks

Identifier	UC-1 Find Nearby Food Trucks
Purpose	Locate nearby food trucks based on current location or manually entered one
Requirements	User Story #1
Development Risks	Ensuring accuracy of location Reliable mapping service for displaying the locations
Pre-conditions	User has the app and is logged in with the app open User must have location services enabled
Post-conditions	User is able to view info about food trucks and go to their location

Table 1: Typical Course of Action

Seq#	Actor's Action	System's Response
1	User opens Food Truck Finder App	System loads and displays the main screen
2	User selects option to find nearby food trucks	System displays a map with markers indicating food trucks
3	User taps on a marker for a food truck for details	System displays information about food truck, such as name, menu, location, displays a button to give directions
	User taps on the option for directions	System opens map and shows a route from the user's location to the chosen food truck
	User arrives at food truck	System detects user reached destination and prompts user to finish the route

Table 2: Alternate Course of Action

Seq#	Actor's Action	System's Response
1	User manually searches for a specific food truck's name	System displays a map with a marker on the location of the food truck
2	User taps on the marker of the food truck for details	System displays information about food truck, such as name, menu, location, displays a button to give directions
3	User taps on the option for directions	System opens map and shows a route from the user's location to the chosen food truck
	User arrives at food truck	System detects user reached destination and prompts user to finish the route

Table 3: Exceptional Course of Action

Seq#	Actor's Action	System's Response
1	User's data is not working and no results are shown on the screen	System cannot print anything to the user and shows an error
2		

3		

System Architecture

(Describe the high-level design of the software.)

Components

10. **Frontend:** Web-based user interface (built with React).
11. **Backend:** Google Maps API (built with Node.js and Express).
12. **Database:** MySQL for data storage.
13. **Authentication:** OAuth 2.0 for user authentication.

Deployment Diagram