Software Requirements Specification

for

GroupEats

Version 2.0

Prepared by

Group Name: Team Red

Kirby Douglas	kdoug020@odu.edu
Matthew Janak	mjana001@gmail.com
Alex Keleher	akele001@odu.edu
Juan Mireles	jmire004@odu.edu
Jajuan Myers	jmyer025@odu.edu
Armando Toledo	atole004@odu.edu

Instructor:	Dr. Ayman El Mesalami
Course:	CS 411W
Teaching Assistant:	Dominik Soos
Date:	November 25, 2024

● Contents

● Revisions

Version	Primary Author(s)	Description of Version	Date Completed
1.1	Matthew Janak	Implement Sections 1& 2	11/6/24
1.2	Kirby Douglas Matthew Janak Alex Keleher Juan Mireles Jajuan Myers Armando Toledo	Implement Section 3	11/7/24
1.3	Alex Keleher	Added Definitions, Assumptions & Dependencies	11/7/24
2.0	Alex Keleher Matthew Janak	Added Dominik's feedback	11/25/24

1 Introduction

1.1 Purpose

GroupEats is a lightweight, responsive web application to plan and execute group dining experiences. A form will collect information such as food preferences, availability, preferred price points, and other relevant details from each user. GroupEats utilizes different APIs and algorithms to create a list of restaurants that best suit the group's overall preferences. (Janak, 2024)

1.2 Scope

This web application will use the PERN stack (PostgreSQL, Express, React, Node.js) to provide an interface where users can set personal dining preferences and view restaurant options from APIs such as Google Places or Yelp. GroupEats will allow group planning by enabling users to access, save, and share restaurant choices that suit everyone's preferences, with secure user accounts and customizable preference. The app will prioritize ease of use, reliability, and a responsive design for various devices. (Janak)

1.3 References

Janak, M. (2024). Lab 1 Draft. CS411W, Dr. Ayman El Mesalami. Version 1, October 2, 2024.

Pettit, R. (n.d.). *Software requirements specification (SRS) template*. George Mason University. Retrieved from https://cs.gmu.edu/~rpettit/files/project/SRS-template.doc

1.4 Definitions, Acronyms and Abbreviations

API (Application Programming Interface): a way for different computer programs or components to interact with one another.

CSS (Cascading Style Sheets): a style sheet language used for describing the presentation of a document written in a markup language like HTML or XML. CSS handles the look and feel of a web page, including layout, colors, fonts, and more.

HTML (Hypertext Markup Language): a markup language used for structuring and presenting content on the World Wide Web.

IDE (Integrated Development Environment): a software application that provides comprehensive facilities to computer programmers for software development.

JSDoc: a markup language used to annotate JavaScript. It allows developers to document their code by adding comments.

JSX (JavaScript XML): a syntax extension for JavaScript that allows developers to write HTML-like markup inside of a JavaScript file.

1.5 Overview

The GroupEats Software Requirements Specification (SRS) defines the functional and non-functional requirements for a web application that facilitates group dining decisions. The document outlines the system's architecture, key features, and integration with third-party APIs such as Yelp to generate restaurant recommendations based on user preferences. The SRS provides a structured foundation for the development, testing, and deployment of GroupEats, ensuring the application meets user expectations for usability, responsiveness, and scalability across web-enabled devices.

2 GroupEats Description

2.1 Product Perspective

GroupEats is a web application that integrates the external API Yelp to aggregate restaurant data and provide users with a tailored selection of dining options. The system's core algorithms analyze input from multiple users, creating a cohesive list of recommendations that meet the group's preferences. The application's modular architecture ensures scalability and compatibility across web browsers.

2.2 Product Functions

Key functions of GroupEats include user profile management, collection of user dining preferences, real-time restaurant data retrieval, and a recommendation system that consolidates group preferences into a single list of options. It also allows users select dining options, and provides notifications when a consensus is reached.

2.3 User Characteristics

The primary users are individuals planning group dining experiences, who are expected to be familiar with basic web navigation. Users are not required to have technical expertise and can interact with the application through a straightforward, guided UI.

2.4 Constraints

N/A

2.5 Assumptions and Dependencies

GroupEats is dependent on the following third-party API:

Yelp API

The application also assumes that users have access to a modern web browser.

3 Specific Requirements

3.1 User Interfaces

The primary UI components include:

- Registration/login screens: allow users to create an account and login
- Preference selection interface: select dining preferences to be shared with the users groups.
- Group management interface: enable users to create and manage dining groups.
- Group event interface: enable users to automatically create a group event based on the preferences of the group.

3.2 Hardware Interfaces

GroupEats is accessible via web browsers on any internet-enabled device. No specialized hardware is required. Supported devices include:

- Personal Computers: Windows, MacOS, Linux
- Tablets: iOS, Android
- Smartphones: iOS, Android

Minimum Hardware Specifications:

- Processor: Dual-core 1.8 GHz or higher.
- Memory: 2 GB RAM or more.

3.3 Software Interfaces

GroupEats interfaces with the following external APIs and technologies:
External APIs:

- Google Places API: retrieves location.
- Yelp API: fetches detailed restaurant information, including location-based restaurant data, restaurant hours, cuisine types, price range, and dietary accommodations.

Technology Stack:

- Frontend: React.js (version 18.x)
- Backend: Node.js (version 16.x) with Express.js (version 4.x)
- Database: PostgreSQL (version 14.x)
- Version Control: GitHub, utilizing continuous integration and deployment.

3.4 Requirements

3.4.1 Functional Requirements

O – Denotes the person who initially drafted the requirement
M_i – Denotes the ith person who contributed to or edited the requirement

3.4.1.1 The system shall display information about the selected restaurant to each member of the group:

The information will only be displayed in the GroupEats application interface.
The date and start time will be displayed.
The restaurant name will be displayed.
The restaurant description pulled from the Yelp API will be displayed including:

- Restaurant Hours
- Price Range
- Cuisine Type
- Dietary Accommodation

O: Toledo

3.4.1.2 Optimal Selection Algorithm

3.4.1.2.1 The system shall make a selection of a time and restaurant for the group in the most optimal way taking into account each of the following factors:

- Each member's availability
- Restaurant hours
- Each member's preferred type of restaurant
  - If too many or too few restaurants meet the criteria, group admin will be able to adjust overall group cuisine preference handling
- Each member's distance from their base location to the restaurant
  - If no times are available, times when most members are available will be displayed

O: Toledo

3.4.1.3 User Selection of Time and Date

3.4.1.3.1 The system shall provide the user the capability to select a time and date for the group outing. O: Mireles

3.4.1.3.1.1 User can select dates and times using an interactive calendar.
3.4.1.3.1.2 The system shall validate selections to prevent scheduling conflicts such as overlapping events or unavailable restaurant hours
3.4.1.3.1.3 If a conflict is detected the system prompts the user with an appropriate error message and suggests an alternate time.

3.4.1.3.2 Users will be able to select a time and date from an interactive calendar, with built in validation to prevent conflicting times. M1: Mireles

3.4.1.4 Group Member Connectivity

3.4.1.4.1 The system must have the ability to connect with other group members. It must have the following function:

- Send an invitation via message or any other electronic messaging system
- Synchronizes group data across all members' devices in real-time.
- Users can view a list of all group members, including their profiles
- Only approved users can access and respond to invitation links. Non-approved users are restricted from accessing the group details.

O: Douglas
M1: Douglas

3.4.1.5 The system must provide the capability for the user to create groups and invite others to join. O: Keleher

3.4.1.6 Group Creation and Management

3.4.1.6.1 Users shall be able to create and name groups for the purpose of planning group dining experiences.

3.4.1.6.1.1 The system shall assign automatically the creator of the group designated as the "Group Admin"

3.4.1.6.1.2 Group admins shall have the ability to remove members from the group

3.4.1.6.1.3 Only the group admin shall be able to complete the following actions:

- Rename group
- Remove members
- Delete group

3.4.1.6.1.4 If the group admin leaves the group the next member with highest longevity will automatically be assigned as the new admin

M1: Keleher
M2: Janak

3.4.1.7 Preference Analysis Algorithm

3.4.1.7.1 The algorithm must be capable of accurately analyzing the preferences of each member of the group. It should take into account individual preferences such as:

- Each member's preferred cuisines
- Accounts for allergies, intolerances and other dietary needs
- Considers the geographical distribution of the group members relative to restaurant locations
- Utilizes historical data on user preferences and previous dining choices

O: Janak

3.4.1.8 The system must provide restaurant suggestions that accurately reflect these preferences. O: Janak

3.4.1.9 The system shall provide the user the capability to enter a unique name for their group. O: Myers

4 Other Non-functional Requirements

5 Other Requirements

6 Appendix A – Data Dictionary

7 Appendix B - Group Log