
Groovy Co.

**Computerized
Food Order and Delivery System
Final Report**

Version 1.0

Revision History

Date	Version	Description	Author
5/14/2024	1.0	Project Team Member	Matthew Munoz
5/14/2024	1.0	Project Team Member	Tyler Ortiz
5/14/2024	1.0	Project Team Member	Joseph Platt
5/14/2024	1.0	Project Team Member	Jonathan Lee

Table of Contents

1.	Introduction	4
2.	System Features	5
3.	Contributions	8
4.	Repository Address	8

Introduction

Groovy Co.'s computerized food order and delivery system uses Node.js, React, and MongoDB. The main prerequisite is that the computer has [Node.js](#) installed. To install the packages required to run the application,

```
npm install
```

must be run in the client and server directories. To run the application,

```
npm run start
```

must be run inside the server directory and then the client directory. The website should then be live at [localhost:3000](#).

Some libraries/dependencies used to implement the application are in no particular order below.

- **Axios**: Axios is a popular HTTP client for making requests to external APIs in a React application. It offers features like promise-based requests, automatic JSON transformation, and interceptors for handling requests and responses globally.
- **Toastify**: React-Toastify is a library for displaying customizable toast notifications in a React application. It provides an easy way to show users alerts, messages, or feedback with minimal setup and configuration.
- **Navigate**: Part of React Router, the Navigate component allows programmatic navigation within a React application, enabling redirects and dynamic routing changes in response to user actions or other events.
- **Zustand**: Zustand is a small, fast, and scalable state management library for React. It provides a simple API for managing and subscribing to state changes without the boilerplate of more complex solutions like Redux.
- **Cookies**: The react-cookie library provides a set of tools for managing browser cookies in a React application. It allows you to easily set, get, and delete cookies to persist user preferences or session data.

System Features

Completed	Partially Implemented
-----------	-----------------------

Actor	Precondition	Actor's Goal	Use Case Identifier	Req Satisfied
All users		Register for CFS as a Chef, Customer, Delivery Person, Importer, or Manager	Register User (UC 1)	1
All users	UC 1	Log in to the system using credentials	Sign In (UC 2)	1
Customer	UC 2	Deposit funds into their account	Deposit Funds (UC 3)	6
Customer	UC 2	Withdraw funds from their account	Withdraw Funds (UC 4)	6
Customer	UC 2	Place food order from list of available options	Order Food (UC 5)	3
Customer	UC 5	Account is frozen if the amount of money in the account is less than the price of the order	Freeze Account (UC 6)	6
Customer	UC 6 & UC 3	Account is unfrozen once the proper amount is deposited (UC 3) into the account	Unfreeze Account (UC 7)	6
Customer	UC 5	Customer chooses to dine in the restaurant	Dine In (UC 8)	3
Customer	UC 8	Customer chooses a time to dine in and chooses preferred sitting (indoor/ outdoor)	Make Reservation (UC 9)	3
Customer	UC 5	Customer chooses to pick up ordered food	Order Pick Up (UC 10)	3
Customer	UC 5	Customer chooses to have food delivered	Order Delivery (UC 11)	3
Customer	UC 8 UC 10	Complain to chef about quality of food	Customer Complain Chef (UC 12)	4
Customer	UC 8 UC 10	Compliment chef on quality of food	Customer Compliment Chef (UC 13)	4
Customer	UC 11	Complain about delivery person or process	Complain Delivery (UC 14)	4
Customer	UC 11	Compliment delivery person or process	Compliment Delivery (UC 15)	4
Customer	UC 5	Leave a public rating of the	Customer Rate Food	1

		food (1-5 stars)	(UC 16)	
Customer	UC 2	Delete account from system	Customer Delete Account (UC 17)	7
Chef	UC 2	Add a dish to the menu. Need a name and price to add to the menu. Description/ tags optional.	Add Dish (UC 18)	8
Chef	UC 18	Remove a dish from the menu	Remove Dish (UC 19)	8
Chef	UC 18	Edit the information of a dish (Name, description, tags, and price)	Edit Dish (UC 20)	8
Chef	UC 2	File complaint about food quality regarding food importer	Complain Importer Quality (UC 21)	9
Chef	UC 3	File complaint about fraud issues regarding food importer	Complain Importer Fraud (UC 22)	9
Deliverer	UC 11	File complaint about customer that was delivered to	Complain Customer (UC 23)	4
Deliverer	UC 11	Compliment a customer that was delivered to	Compliment Customer (UC 24)	4
Manager	UC 12 UC 14 UC 21 UC 22 UC 23	Process a complaint and see if it is valid	Process Complaint (UC 25)	4
Manager	UC 25	Convert a valid complaint against a user in to a formal warning or give formal warning to users that filed complaint without merit	Give Formal Warning (UC 26)	4
Manager	UC 13 UC 15	Promote an employee after two consecutive compliments are recieved by customers	Promote Employee (UC 27)	2
Manager	UC 26	Demote an employee after two consecutive warnings are recieved or if chef files	Demote Employee (UC 28)	2

		an invalid complaint against an importer		
Manager	UC 28 (UC 22 & UC 25)	Fire an employee after two demotions are recieved or if importer recieves a valid fraud complaint	Fire Employee (UC 29)	2
Manager	UC 17 UC 26	Close a users account and refunds any money in account if the customer deletes their or if non-VIP user has two formal warnings against them	Close Customer Account (UC 30)	5, 6
Manager	UC 24	Grant a user VIP status if two consecutive compliments are recieved	Grant VIP (UC 31)	5
Manager	UC 26	Take a user's VIP status and revert to a regular user with warning cleared	Take VIP (UC 32)	5
Chef, Customer, Deliverer, or Importer	UC 12 UC 14 UC 21 UC 23	Dispute a complaint made by another user	Dispute Complaint (UC 33)	4, 9
Importer	UC 2	Add ingredient to the inventory	Add Ingredient (UC 34)	10
Chef or Manager	UC 34	Remove ingredient from the inventory	Remove Ingredient (UC 35)	10

Contributions

There are no concerns among the team; all members provide their work on various aspects of the projects. Since the last report updates and work have been as follows:

- Tyler Ortiz worked on ingredient management, fulfillment, and manager systems
- Matthew Munoz worked on accounts, landing pages, and menu
- Joseph Platt provided the pseudocode for all methods
- Jonathan Lee worked on debugging and the final report

Tyler Ortiz and Matthew Munoz were exceptional in their contributions to this project, and all the members agree they should get a bonus of 7%.

Repository Address

<https://github.com/MattMunoz/Groovy-Co>