# Stakeholder List

## Primary Stakeholders

| Stakeholder | Role | Interest |
|---|---|---|
| Customers | Customers make purchases on the app. Primary consumer. | Customers want the app to be able to order food and have it ready by the time they arrive at the dining establishment. |
| Staff | Staff actually create the food that has been ordered on the app. | The staff want the app to work properly as it may come out of their pay if they make mistakes or upset any users. |
| Admin | Admins facilitate a smooth flow of data between customers and staff. | An app that allows customers to easily place orders at their location would potentially cause profits to increase, possibly giving the admin reputation with their higher ups. |

## Secondary Stakeholders

| Stakeholder | Role | Interest |
|---|---|---|
| Lab TAs | Lab TAs ensure that the product is up to the given standards. | The Lab TAs are in charge of answering questions the developers may have, so if an entire lab of developers comes out with spectacular apps, the professor knows that the TAs did a good job of answering questions and addressing concerns. |
| Instructors | Instructors are the people that come up with the idea in the first place, and give developers a task. | It is the instructor’s job to provide the initial information for developers to create the app, both the information relevant to software, and the software development process. If the majority of their students do not do well on the project, then it reflects poorly on their ability. |
| Development Team | The development team ensures the software is fully functional and meets the requirements | It is the development team's job to ensure the app works smoothly for all primary stakeholders: customers, staff, and admins. If the development team does not maintain the use of the |

| | | |
|---|---|---|
| | | application, the application will be inefficient or unusable |

### Tertiary Stakeholders

| Stakeholder | Role | Interest |
|---|---|---|
| Future Developers | Future developers will continue development after the current development team has passed on the project | Future developers need to understand the current implementation and testing of the application to ease development later on. If there is mismatched code or poor documentation, development may be hard and potentially break the application. |

## Stakeholder Biases

### Primary Stakeholders

| Stakeholder | Bias | Why? |
|---|---|---|
| Customers | Status quo bias (aversion to change) | It's common for customers to continue ordering food in the manner that they are used to, whether that be in person, through food delivery services, phone, etc. They might resist adopting the app if it feels unfamiliar or inconvenient to what they've become acclimated to, preventing a transition into this new system. |
| Staff | Self-serving/Self-esteem bias | Rather than own up to mistakes in the app, staff may just blame the app altogether for certain errors, because it can have an adverse outcome on their livelihood/esteem. This will also in turn affect the way they perceive certain changes in the system. |
| Admin | Authority bias | Admins might make assumptions on certain higher-ups' beliefs without fully knowing the actual reality of what they want, which can ultimately influence how the system works/is |

| | | |
|---|---|---|
| | | managed, along with which features are prioritized. |

### Secondary Stakeholders

| Stakeholder | Bias | Why? |
|---|---|---|
| Lab TAs | Confirmation Bias | When grading or evaluating code, Lab TAs could seek validation from instructors for even minor decisions, which could impact decisions on grading further down the road |
| Instructors | Sunk Cost Fallacy | As this project could be used for multiple semesters, developers could use code or ideas from previous developers, making the learning objective |
| Development Team | Planning Fallacy | The development team may underestimate the amount of time it takes to deliver a milestone, which could result in a late turn-in or poorly written code |

### Tertiary Stakeholders

| Stakeholder | Bias | Why? |
|---|---|---|
| Future Developers | Hard-Easy Effect | Future developers could go into the project thinking it is a simple project and underestimate the complexity of the implementation, leading to confusion and inefficient development |

# Prompt Crafting

## Zero-Shot Prompting

Zero-shot prompting is when a user gives a LLM a prompt with no examples, only instructions. Using this prompting method, the knowledge already in the LLM makes its own assumptions of how the task is completed, since there is no example basis.

## Careful Prompting

Careful prompting is when a user gives a LLM a prompt that has been thoughtfully written which can include examples, specific word choice, and specific structure of the prompt. This gives the LLM more instructions to follow, leading it by example rather than its own generation from its knowledge

## Prompting Comparison

| Features | Zero-Shot Prompting | Careful Prompting |
|---|---|---|
| Examples Provided | No | Yes |
| Goal | Utilize LLM's pre-trained knowledge | Maximize intention, precision, control over output |
| Strengths | Fast and easy to input | Higher accuracy and consistency in output |
| Weaknesses | Possible vague and inaccurate output | More effort and awareness to craft prompt |
| Best Use Cases | Broad tasks with low expectations; quick prototyping, general questions | High-stakes, intense tasks of exact output; Technical writing, workflow automation |

# Use Cases

## UC 1: Add a New Item to Inventory

| Pre-reqs | Short Description | Main Flow | Alternative Flows |
|---|---|---|---|
| The user (Staff or Admin) is | Staff or admin can add a new | 1.User logs into system<br>2. User clicks "Add Item" button<br>3. User fills out the required information | **[Invalid Input]**: One or more input fields are empty |

| | | | |
|---|---|---|---|
| logged into the system.<br><br>The user has the appropriate role-based access (Staff or Admin). | item to inventory | including name, description, and amount<br>4. User clicks the “Submit” button [Invalid Input] [Duplicate Name]<br>5. The new item is added to the inventory | **[Duplicate Name]**: Name of new item already exists in inventory |

## UC 2: Delete an Item in Inventory

| Pre-reqs | Short Description | Main Flow | Alternative Flows |
|---|---|---|---|
| The user (Staff or Admin) is logged into the system.<br><br>The user has the appropriate role-based access (Staff or Admin). | Staff or admin can add an item in inventory | 1.User logs into system<br>2. User clicks “Delete Item” button<br>3. User searches for the name of item to be deleted in dropdown menu and selects it<br>4. User clicks the “Submit” button [Cannot Delete]<br>5. The item is removed from inventory | **[Cannot Delete]**: Item to be delete exists in at least one recipe and cannot be deleted |

## UC 3: Create A Recipe

| Pre-reqs | Short Description | Main Flow | Alternative Flows |
|---|---|---|---|
| The user (Staff or Admin) is logged into the system. | Staff or admin can create a new recipe from items in inventory | 1.User logs into system<br>2. User clicks “Create Recipe” button<br>3. User fills out the required information | **[Invalid Input]**: One or more input fields are empty or have an invalid value<br><br>**[Duplicate Name]**: Name of new |

| | | | |
|---|---|---|---|
| The user has the appropriate role-based access (Staff or Admin). | | including name, description, items from inventory, and cost<br>4. User clicks the “Submit” button [Invalid Input] [Duplicate Name]<br>5. The new recipe is added to the menu | recipe already exists on the menu |

## UC 4: Delete a Recipe

| Pre-reqs | Short Description | Main Flow | Alternative Flows |
|---|---|---|---|
| The user (Staff or Admin) is logged into the system.<br><br>The user has the appropriate role-based access (Staff or Admin). | Staff or admin can delete a recipe from the menu | 1.User logs into system<br>2. User clicks “Delete Recipe” button<br>3. User searches for the recipe name to be deleted in dropdown menu and selects it<br>4. User clicks the “Submit” button [Cannot Delete]<br>5. The recipe is removed from the menu | **[Cannot Delete]**: Recipe has a current order in line for a customer and cannot be deleted until order is fulfilled |

## UC 5: Edit a Recipe

| Pre-reqs | Short Description | Main Flow | Alternative Flows |
|---|---|---|---|
| The user (Staff or Admin) is logged into the system.<br><br>The user has the appropriate | Staff or admin can edit an recipe from the menu | 1.User logs into system<br>2. User clicks “Edit Recipe” button<br>3. User searches for the recipe name to be edited in dropdown menu and selects it<br>4. User edits parameters in question such as name, | **[Invalid Input]**: One or more input fields are empty or have an invalid value<br><br>**[Duplicate Name]**: Name of edited recipe already exists on the menu |

| | | | |
|---|---|---|---|
| role-based access (Staff or Admin). | | item counts, price, etc.<br>4. User clicks the “Submit” button [Invalid Input] [Duplicate Name]<br>5. The recipe is updated in the menu | |

## UC 6: Manage Staff & Customers

| Pre-reqs | Short Description | Main Flow | Alternative Flows |
|---|---|---|---|
| The user (Admin) is logged into the system.<br><br>The user has the appropriate role-based access (Admin). | The admin needs to be able to create/edit/ delete staff/customer user accounts | 1. An admin selects the "Manage Staff and Customers" option.<br>2. The system displays a list of all existing staff and customer users.<br>3. The admin can choose to:<br>a. Create a New User:<br>i. The admin selects the "Add New User" option.<br>ii. The system prompts the admin to enter details (e.g., username, password, role, contact information). [Invalid Input]<br>iii. The admin enters the information and submits it.<br>iv. The system validates the information and saves the new user, displaying a confirmation message.<br>b. Edit an Existing User:<br>i. The admin selects a user from the list to edit.<br>ii. The system displays the current details of the selected user.<br>iii. The admin modifies the necessary fields and submits the changes. [Invalid Input]<br>iv. The system displays a | [Invalid Input]: If the admin enters invalid information (e.g., missing required fields) while creating or editing a user, the system displays an error message indicating the required fields and prompts the admin to correct the input. |

| | | | |
|---|---|---|---|
| | | confirmation message.<br>c. Delete a User: | |

## UC 7: Customer Orders

| Pre-reqs | Short Description | Main Flow | Alternative Flows |
|---|---|---|---|
| The user (Customer) is logged into the system.<br><br>The user has the appropriate role-based access (Customer). | Customers need to be able to add items to their order and then make a purchase while leaving a tip. | 1. A customer selects the Order Items option from the menu.<br>2. The system displays a list of available recipes/items.<br>3. The customer browses the items and selects one or more recipes/items to add to their order.<br>4. The customer can specify the quantity for each selected item.<br>5. The system calculates the order total, including the selected items, applicable sales tax, and displays it to the customer.<br>6. The customer is prompted to enter a tip, choosing from options of 15%, 20%, 25%, or a custom amount.<br>7. The customer reviews the order summary and confirms the order. | [Order Time Limit]:<br>If the customer takes too long to finalize the order, the system times out and prompts them to restart the ordering process. |

| | | | |
|---|---|---|---|
| | | [Order time out]<br>8. The system processes the order and displays a confirmation message with the order details.<br>9. The customer is returned to the order menu or home screen. | |

## UC 8: View & Fulfill Orders

| Pre-reqs | Short Description | Main Flow | Alternative Flows |
|---|---|---|---|
| Staff is logged into the system. | Staff users need to be able to view orders and fulfill them. | 1. Staff can look at current orders waiting to be fulfilled.<br>2. Staff members can choose which order they would like to fulfill.<br>3. Staff member chooses to fulfill an order [Order fulfilled by other Staff].<br>4. The system updates the order to show that it is fulfilled.<br>5. The person who ordered is notified that their order has been fulfilled by a staff member and is ready for pick up [UC 5]. | [Order fulfilled by other Staff]:<br>If when a staff member tries to fulfill an order the order has already been fulfilled by another staff member the staff member is displayed an error message stating that the order has already been fulfilled and the page is reloaded to show current orders waiting to be filled. |

## UC 9: Pickup Notification

| Actors | Short Description | Main Flow | Alternative Flows |
|---|---|---|---|
| Customers and staff members are logged into the system.<br><br>Customer places an order [UC #3] and a staff member fulfills that order [UC #4]. | Customers need to be able to see when their order is fulfilled so they can pick it up. | 1. Customer's screen displays a notification saying that their order is ready for pickup.<br>2. The system logs information and updates the customer's view of their order even after the notification goes away.<br>3. Customer picks up their order and confirms in the system that they received it [No Confirmation]. | [No Confirmation]:<br>If the customer fails to complete the confirmation of pickup, then after 2 hours, the system will log the order as confirmed anyways since the customer did not pick up their order in time. |

## UC 10: Set Sales Tax Rate

| Pre-reqs | Short Description | Main Flow | Alternative Flows |
|---|---|---|---|
| Admin is logged into the system. | Admin users need to be able to set a sales tax rate for the system. | 1. Admin selects the option to set the sales tax rate.<br>2. Admin enters the new sales tax rate.<br>3. The system validates the sales tax rate [Invalid Rate].<br>4. The system updates the sales tax rate | [Invalid Rate]:<br>If the input sales tax rate is not a valid positive decimal number (ex. greater than or equal to 0 and less than or equal to 100), an error message is displayed, and the Admin is returned to the form to enter a valid rate. The tax rate isn't updated. |

| | | | |
|---|---|---|---|
| | | and logs the update with Admin details.<br>5. Admin is returned to the main menu. | |