Project 1a1
By: Sagnik, Aadhir, Frank, Jackson

1) List of stakeholders

1. **Customers** – Students, faculty, and visitors placing orders.

2. **Staff (Baristas/Order Fulfillers)** – Prepare and complete customer orders.

3. **Staff (Cashiers/Front-of-House)** – Handle customer interactions, edits, and payments.

4. **Administrators** – Manage roles, tax rates, staff accounts, and inventory.

5. **Campus IT / Identity Services** – Provide authentication and single sign-on integration.

6. **Privacy & Security Office** – Ensure compliance, data protection, and secure role-based access.

7. **Accessibility Office** – Enforce WCAG standards and usability for all users.

8. **Dining Services / Operations Management** – Oversee real-world workflow, inventory, and order fulfillment policies.

9. **Payment Processor / Finance (Tax Authority)** – Handle transactions, tips, and apply NC's 2% sales tax.

10. **Customer Support Team** – Handles customer inquiries, complaints, and order issues; provides feedback on system usability and policies for improving customer experience.

2. **Stakeholder biases**

Stakeholders Involved	**Potential Conflict / Clash**	**Reason / Explanation**
Staff (Baristas) vs Dining Services	Flexible/custom orders vs. standardized workflow	Customers request customizations; Operations prioritize efficiency and predictable prep.
Accessibility Office vs Customers	Strict accessibility rules vs. flashy UI/animations	Accessibility standards may restrict animations, color schemes, or tab ordering that enhance visual experience.

Administrators vs Customers	Strict role-based access vs. seamless guest checkout	Admins prioritize control and auditability; customers prefer frictionless login-free ordering.
Customer Support vs Privacy Office	Detailed order history vs. minimal stored data	Support needs past order info to resolve complaints; Privacy wants to minimize stored personal data.
Campus IT / Identity Services vs Customers	Strong authentication vs. convenience	IT enforces SSO/security policies; customers want one-click or quick access.

3) **Zero shot prompting vs careful prompting**

The main difference between these two methods is how much context and help is given to the AI to get the right answer.

One-shot prompting often involves giving the AI a single example. It basically involves showing the AI one problem, giving it the correct answer, and then giving it a very similar problem to then solve It's a quick and easy way to tell the AI, "Do it like this." This works great for simple stuff, like changing the format of some text or translating a basic sentence and is also done very quickly. The downside is that since it only has one example to go on, it can get confused if the new problem is too different or tricky, often providing incorrect or undesirable outputs.

Careful prompting involves giving the AI a clearer background and more context. Instead of just one example, you oftentimes give it a few. You also break down your instructions step-by-step and explain how to think through the problem. This is the way to go for complicated questions, like coding problems or anything that requires logic. While this process takes more time to set up, it helps the AI avoid mistakes and gives you a much more accurate and detailed answer.

So, if the task is easy and straight-forward and need a quick answer, use one-shot. If it's hard and you need the answer to be perfect, take the time to use careful prompting.

4) **Use cases**

UC1: Customer Places an Order

Preconditions: Customer is logged in or anonymous checkout is enabled.
Main Flow:

1. Customer selects items to order.
2. System displays order summary with prices and tax.
3. Customer selects tip percentage or enters a custom tip.
4. Customer confirms the order.
5. System generates the order and assigns it to staff.

Subflows:

- [Select Items] Customer adds multiple items, including multiples of the same item.
- [Confirm Order] If payment fails, customer is prompted to retry payment.

Alternative Flows:

- [Payment Failure] System notifies customer of payment failure and allows retry.
- [Empty Cart] System prevents order confirmation if no items are selected.

UC2: Staff Fulfills an Order

Preconditions: Staff member is logged in with fulfillment permissions.
Main Flow:

1. Staff views list of pending orders.
2. Staff selects an order to fulfill.
3. System marks the order as in progress.
4. Staff completes preparation.
5. System marks order as ready for pickup.

Subflows:

- [Select Order] Staff can filter by order time or priority.
- [Mark Ready] Staff confirms items are complete before updating status.

Alternative Flows:

- [Incorrect Item] Staff can edit the order if a mistake occurred.

UC3: Customer Picks Up Order

Preconditions: Customer has an order marked ready for pickup.
Main Flow:

1. System notifies customer that the order is ready.
2. Customer arrives at pickup location.
3. Staff confirms order pickup.
4. System marks order as completed.

Subflows:

- [Notify Customer] Notification via app, email, or SMS.
- [Pickup Verification] Staff scans order ID or confirms customer identity.

Alternative Flows:

- [Order Not Ready] System informs customer if the order is not yet ready.

UC4: Administrator Adds New Staff Member

Preconditions: Administrator is logged in with user management permissions.
Main Flow:

1. Admin selects “Add Staff.”
2. Admin enters staff details and role.
3. System validates data and creates staff account.

Subflows:

- [Validate Data] System checks for missing or duplicate info.
- [Assign Role] Admin selects permissions for staff.

Alternative Flows:

- [Invalid Input] System rejects incomplete or invalid input and prompts admin to correct.

UC5: Administrator Edits Customer Profile

Preconditions: Administrator is logged in.
Main Flow:

1. Admin searches for the customer.
2. Admin selects the customer profile.
3. Admin edits necessary details (e.g., contact info).
4. System saves changes and confirms update.

Subflows:

- [Search Customer] Admin can filter by name, email, or ID.

Alternative Flows:

- [Customer Not Found] System displays error and suggests retry.

UC6: Administrator Sets Sales Tax Rate

Preconditions: Admin is logged in.
Main Flow:

1. Admin navigates to tax settings.
2. Admin enters new tax rate.
3. System validates rate and updates pricing calculations.

Subflows:

- [Validate Rate] System ensures rate is within legal limits.

Alternative Flows:

- [Invalid Rate] System rejects negative or non-numeric input.

UC7: Customer Requests Anonymous Order

Preconditions: Anonymous checkout is enabled.
Main Flow:

1. Customer selects items and proceeds to checkout without logging in.
2. System collects minimal necessary info (pickup code, payment).
3. Customer confirms order.
4. System generates order and assigns to staff.

Subflows:

- [Assign Pickup Code] System generates temporary order identifier.

Alternative Flows:

- [Payment Failure] System prompts for retry or cancellation.

UC8: Staff Views Order History

Preconditions: Staff member is logged in.
Main Flow:

1. Staff navigates to order history.
2. System displays completed and past orders.
3. Staff can filter orders by date, customer, or status.

Subflows:

- [Filter Orders] Staff can generate reports for inventory or performance.

Alternative Flows:

- [No Orders Found] System displays message if no orders match filters.

UC9: Customer Leaves a Tip

Preconditions: Customer has items in cart and is at checkout.
Main Flow:

1. Customer selects tip amount (15%, 20%, 25%, custom).
2. System calculates total with tip included.

3. Customer confirms payment.
4. System updates order with tip details.

Subflows:

- [Custom Tip] Customer enters manual tip amount.

Alternative Flows:

- [Negative Tip] System prevents negative or invalid input.

UC10: Customer Receives Order Notification

Preconditions: Customer has an active order.
Main Flow:

1. System tracks order status.
2. When order is ready, system sends notification (app, email, SMS).
3. Customer is informed of pickup location and order ID.

Subflows:

- [Notification Channels] System chooses available channel(s) for customer.

Alternative Flows:

- [Notification Failure] System retries sending or displays warning in app.