

The new use cases (≥10, 5 pages at least)

1. UC-1: Customer Places Order

Preconditions:

- Customer logged in (or guest ordering allowed).
- Menu items exist and are available in inventory.

Main Flow:

1. Customer browses menu.
2. Selects one or more items.
3. Adds them to cart.
4. Proceeds to checkout.
5. System displays subtotal, tax, tip options.
6. Customer selects tip and confirms payment.
7. System processes payment.
8. System creates order and adds it to queue.
9. Customer receives confirmation.

Sub Flow:

- [Tip Selection]: Choose suggested/custom tip.
- [Tax Calculation]: Apply current tax rate from admin settings.

Alternative Flow:

- Payment failed → Order not created, retry option.
- Item out of stock → Removed, prompt to adjust.
- Cancel before payment → Cart cleared, no order.

2. Browse Menus & Restaurants

Preconditions:

- Customer logged in.
- Restaurants and menus available in system.

Main Flow:

1. Customer opens restaurant listing.
2. Applies filters (cuisine, rating, location).
3. Selects restaurant.
4. System loads menu with items.

Sub Flow:

- User views descriptions, prices, images.
- User checks ratings/reviews.

Alternative Flow:

- No restaurants → Show fallback message.
- Menu item unavailable → Notify with alternatives.

3. Process Payments via Gateway

Preconditions:

- Customer has items in cart.
- Valid payment method exists.

- Payment gateway is active.

Main Flow:

1. Customer confirms checkout.
2. System sends request to gateway.
3. Gateway processes and confirms.
4. System records transaction.

Sub Flow:

- Promo code validation (if applied).

Alternative Flow:

- Payment failure → Retry/alternate method.
 - Gateway timeout → Queue and retry.
-

4. UC-6: View Order History (Customer)

Preconditions:

- Customer logged in.

Main Flow:

1. Customer navigates to “My Orders.”
2. System fetches past orders.
3. Orders displayed with details (items, total, tips, taxes).

Sub Flow:

- [Repeat Order]: Customer selects previous order to reorder (future extension).

Alternative Flow:

- No order history → Show empty state.
-

5. UC-7: Customer Picks Up Order

Preconditions:

- Order fulfilled by staff.
- Customer waiting for notification.

Main Flow:

1. Customer notified "Order Ready."
2. Customer arrives at pickup counter.
3. Staff hands order to customer.
4. Customer/staff confirms pickup.
5. System marks order "Completed."

Sub Flow:

- Staff scans order code/ID for verification.

Alternative Flow:

- Delayed pickup → Order remains "Fulfilled."
 - Wrong customer attempts pickup → Staff verifies ID.
-

6. UC-2: Staff Views and Fulfills Orders

Preconditions:

- Staff logged in.
- Pending orders exist in system.

Main Flow:

1. Staff navigates to “Orders.”
2. System shows pending queue.
3. Staff selects order.
4. Prepares order.
5. Marks as “Fulfilled.”
6. System updates status.

Sub Flow:

- Filter orders by urgency, time, or size.

Alternative Flow:

- Item unavailable → Cancel order, notify customer.
- Staff error → Revert fulfillment status.

7. UC-4: Staff Manages Inventory

Preconditions:

- Staff logged in.

Main Flow:

1. Staff navigates to “Inventory.”
2. Adds/updates items or stock.

3. System validates and updates records.

Sub Flow:

- Add new item with details.
- Update quantity for existing item.

Alternative Flow:

- Invalid input (negative, non-numeric) → Rejected.
 - Duplicate item → Merge/update.
-

8. Manage Menu & Pricing

Preconditions:

- Restaurant staff logged in.

Main Flow:

1. Staff navigates to “Menu Management.”
2. Adds, edits, deletes menu items.
3. System saves and updates customer app.

Sub Flow:

- Upload/edit item images.
- Adjust prices.

Alternative Flow:

- Invalid input → Prompt correction.
- System downtime → Queue changes.

9. UC-5: Admin Sets Tax Rate

Preconditions:

- Admin logged in.

Main Flow:

1. Admin navigates to “Settings → Tax.”
2. Inputs new tax %.
3. System validates and saves.
4. Future orders use new rate.

Sub Flow:

- Admin reviews current rates before editing.

Alternative Flow:

- Invalid rate → Rejected.
- Cancel → No changes saved.

10. UC-9: System Sends Order Notifications

Preconditions:

- User (customer/staff) has notifications enabled.

Main Flow:

1. Event triggers (order placed, ready, cancelled).
2. System generates notification.

3. Notification delivered via in-app, SMS, or email.

Sub Flow:

- Templates: “Order Ready,” “Cancelled,” “Delayed.”

Alternative Flow:

- Delivery failure → Retry or fallback channel.

Reflection document:

1. How did you decide what NOT to do?

The decision of what not to build centered on validating the core value loop, browse, order, pay, deliver, with the least effort, so non-essential, high-complexity, or scale-dependent features were explicitly excluded using structured prioritization and value-versus-effort thinking to maximize validated learning per unit of work while minimizing technical and operational risk.

What to exclude:

- Ratings/reviews, promo codes, subscriptions, emergency prioritization: require network effects, marketing engines, complex policy logic, and do not validate core fulfillment loop in v1.
- Driver accept/reject and optimized routing: autonomy and routing optimization add complexity and dependency on traffic data; initial assignment plus external navigation is sufficient to validate delivery reliability.
- Inventory scanning, advanced settlements, dispute automation, analytics suites, fraud scoring: high integration and process overhead better deferred until volume warrants it and baseline processes stabilize.
- Multi-channel notifications and broad third-party integrations: begin with one or two reliable channels and minimum viable API set to reduce surface area and failure modes.

Why exclusions are justified for an MVP:

- MVP success hinges on verifying that users can discover, order, and receive food on time with minimal friction; anything beyond that risks delaying the primary learning loop and inflating burn.
- Lean delivery in food logistics rewards early operational validation over feature breadth; early shipping secures feedback and iteration speed in a competitive category.

2. What negative impacts or disappointments this MVP could have for your stakeholders.

Some customers may want more payment options like apple pay or paypal but our MVP only supports credit and debit cards. Secondly we do not have more options for delivering food to the customers at their home like uber eats, doordash etc, but MVP only supports pickup for the most part. There is no real time tracking available of the order and the system does not tell the customer the estimated time in which the order

may be ready/delivered which may feel less modern as compared to competitors, since now most of the online ordering apps have it. There are no royalty points or discount features available as well to please the customers. There is not a lot of automation in the MVP for example, the staff can only add/update stock but not bulk upload or auto-sync. There are no sales reports, or dashboards to get insights to have performance tracking. Customers can also not leave reviews for the food which they like or dislike so there will be no way for the staff to know which food items should be removed or kept in the menu, they would just have to manually check which food item was ordered the least by customers. The UI also seems to be very basic for a cafe, since it needs more of the feature enhancements mentioned above.

3. What changes you made (and why) to the MVP to appease at least some of the stakeholders.

1. Customers

Clear Tax/Tip Display: UC-1 to show subtotal, tax calculation and tip options (15%, 20%, 25% or custom) before payment to address transparency concerns.

Order Status Updates: UC-10 via email/SMS for “Order Ready” status and detailed order history (UC-4) to compensate for lack of real-time tracking.

Guest Checkout: Added option to order without registration to reduce friction (Data security concern/ Quick first order) for first-time users.

2. Staff (Baristas/Kitchen Workers)

Efficient Order Queue: Staff dashboard (UC-6) with filtering by urgency, time and size to manage busy periods according to the manager.

Quick Inventory Updates: UP inventory management (UC-7) for fast add/update of stock levels to prevent bottlenecks from out of stock items.

Clear Order Details: Queue displays full item lists and special instructions to ensure accurate preparation.

3. Administrators

Payment Security: Gateway integration (UC-3) with error handling and retry logic to maintain transaction reliability.

Tax Compliance: UC-9 Dedicated tax configuration with validation to ensure NC sales tax compliance across all orders.

4. Prompt history:

CLAUDE [FINAL 10 USE CASES WITH JUSTIFICATIONS ON REJECTED ONES]:

<https://claude.ai/share/36ca2fa6-8424-411f-ba36-ed79090baf7b>

CHATGPT [FINAL 10 USE CASES WITH JUSTIFICATIONS ON REJECTED ONES]:

<https://chatgpt.com/share/68c36c44-1058-8001-b056-266ef4e85520>

GEMINI [FINAL 10 USE CASES WITH JUSTIFICATIONS ON REJECTED ONES]:

<https://www.perplexity.ai/search/i-ll-give-you-a-pdf-containing-S5YMYxTWQ4aRC1PSvPTErA>