

hello. lets work on the next part of the project.

Time to get serious, and minimal.

LLMs give you many choices. Which will you use?

Your goal is a **MVP (minimal viable product)**: the most basic version of a product that allows a team to collect the maximum amount of validated learning about customers with the least amount of effort.

Decide what from 1b1 you will **NOT** be doing, and justify why you will not be doing that.

For this task, keep showing the two LLMs the current use cases and ask them for answers to part 4 (in this list).

Generate at least 10 new use cases (5 pages total) describing the new design, each with:

- Preconditions
- Main Flow
- Subflows
- Alternative Flows

We're sticking with a single, capable LLM for our MVP. This is the **most basic version** of the product, as using just one model allows us to collect the **maximum amount of validated learning** about customer needs with the **least amount of effort**. The complexity and resource overhead of managing, training, and integrating multiple LLMs would directly oppose the core principles of an MVP. 🚀

What We're Not Doing (From 1b1)

Based on the **MVP** philosophy, we will not be implementing several features that were likely part of the initial, more complex design (1b1). These features, while potentially valuable later on, add unnecessary development effort and are not essential for gathering initial customer feedback.

- **Advanced Analytics and Reporting:** A full-fledged analytics dashboard is not needed for the MVP. Instead, we'll focus on simple, core metrics like the number of successful interactions and common user queries. This allows us to quickly validate the product's fundamental utility without spending time building complex reporting tools.
- **Third-Party Integrations:** We're holding off on integrating with external services like Slack, Trello, or Jira. While a future goal, building these integrations would require

significant effort and could introduce points of failure. The MVP will function as a standalone tool to prove its value first.

- **Customizable User Interfaces (UIs):** The MVP will use a standardized, minimal UI. Allowing users to customize themes, layouts, or shortcuts adds complexity that isn't necessary for the initial testing phase. A simple, functional interface is sufficient to prove the product's core value proposition.

By cutting these features, we can accelerate our development timeline and focus all our effort on perfecting the core interaction between the user and the LLM.

New Use Cases for the MVP

Use Case 1: Generate a Project Plan from a High-Level Goal

- **Preconditions:** The user is logged into the application and has a high-level project goal in mind.
 - **Main Flow:**
 - The user types a request into the chat interface, e.g., "Generate a project plan for a new marketing campaign."
 - The LLM prompts the user for specific details, such as the campaign's target audience and key deliverables.
 - The user provides the requested information.
 - The LLM generates a detailed, step-by-step project plan, including phases, tasks, and estimated timelines.
 - The LLM presents the plan to the user in a clear, formatted output.
 - **Subflows:**
 - **Refining the plan:** The user can request specific changes, like "Make the 'content creation' phase longer." The LLM updates the plan accordingly.
 - **Alternative Flows:**
 - **Insufficient Detail:** If the user's initial request is too vague, the LLM will explain what additional information is needed to proceed.
 - **Invalid Request:** If the user asks for something unrelated to project planning, the LLM will inform them of its purpose and redirect them to a valid query.
-

Use Case 2: Summarize a Long Document

- **Preconditions:** The user has a long text document (e.g., a meeting transcript, a report, a research paper) ready to be summarized.
- **Main Flow:**
 - The user pastes the document into the application or uploads a text file.

- The user requests a summary, specifying a desired length (e.g., "Summarize this in three bullet points").
 - The LLM processes the document.
 - The LLM generates a concise summary based on the user's criteria.
 - The summary is displayed to the user.
 - **Subflows:**
 - **Highlighting key terms:** The user can ask for a list of the most important keywords or phrases from the document.
 - **Alternative Flows:**
 - **Unsupported File Type:** If the user attempts to upload a non-text file (e.g., a PDF or an image), the LLM will alert them that the file type is not supported.
 - **Document Too Long:** If the document exceeds the processing limit, the LLM will inform the user and suggest they try a shorter excerpt.
-

Use Case 3: Create a Stakeholder Communication Email

- **Preconditions:** The user needs to draft a professional email to update stakeholders on a project's status.
 - **Main Flow:**
 - The user provides key talking points or bullet points they want to include in the email.
 - The user specifies the email's tone (e.g., "formal," "informal," "urgent").
 - The LLM generates a complete, well-structured email draft using the provided points and tone.
 - The user can copy the email directly from the interface.
 - **Subflows:**
 - **Changing the tone:** The user can ask the LLM to rewrite the email in a different tone, such as "Now make it more direct and to the point."
 - **Alternative Flows:**
 - **Missing Information:** If the user's initial input is too sparse, the LLM will ask for more context, such as the recipient's name or the project's current status.
 - **Unclear Instructions:** If the user's request is ambiguous, the LLM will offer a few different email drafting options and ask the user to clarify their intent.
-

Use Case 4: Brainstorm Ideas for a New Feature

- **Preconditions:** The user is working on a new product or feature and needs creative input.
- **Main Flow:**
 - The user describes the product and the problem they are trying to solve.

- The LLM asks a series of clarifying questions to better understand the user's needs.
 - Based on the conversation, the LLM generates a list of creative ideas for the new feature.
 - The user can review the list and ask for more details on any specific idea.
 - **Subflows:**
 - **Expanding on an idea:** The user can ask, "Tell me more about idea number 3," and the LLM will provide a detailed description of that concept.
 - **Alternative Flows:**
 - **Unrealistic Scope:** If the user's goal seems too broad or unrealistic, the LLM will gently suggest narrowing the focus or breaking the problem down into smaller, more manageable parts.
-

Use Case 5: Generate Meeting Agenda

- **Preconditions:** The user needs to prepare a structured agenda for an upcoming meeting.
 - **Main Flow:**
 - The user provides the meeting's topic and a few key objectives.
 - The user specifies the meeting's duration.
 - The LLM creates a timed agenda with suggested discussion points and time allotments for each.
 - The agenda is displayed in a clear, formatted list.
 - **Subflows:**
 - **Reordering items:** The user can ask the LLM to move a specific item to a different point in the agenda.
 - **Alternative Flows:**
 - **Duration Mismatch:** If the user provides too many topics for the given duration, the LLM will suggest either a longer meeting or prioritizing the most important discussion points.
-

Use Case 6: Write a Job Description

- **Preconditions:** A user needs to create a job description for a new role.
- **Main Flow:**
 - The user provides the job title and a brief description of the required duties.
 - The LLM asks for more details, such as required skills, experience level, and key responsibilities.
 - The user provides the requested information.
 - The LLM generates a complete, professional job description.
 - The user can copy the text.

- **Subflows:**
 - **Adding a specific section:** The user can ask to add a "company culture" or "benefits" section to the description.
 - **Alternative Flows:**
 - **Incomplete Information:** If the user's initial input is vague, the LLM will provide a list of common responsibilities for the given job title and ask the user to select the most relevant ones.
-

Use Case 7: Create a Simple Marketing Slogan

- **Preconditions:** A user needs a catchy slogan or tagline for a product or company.
 - **Main Flow:**
 - The user describes the product or company and its core values.
 - The LLM generates a list of several slogan options.
 - The user reviews the list and can ask for more variations.
 - **Subflows:**
 - **Slogans with a specific tone:** The user can ask for slogans with a more "humorous" or "inspiring" feel.
 - **Alternative Flows:**
 - **Redundant Input:** If the user provides an overly complex or redundant description, the LLM will ask them to simplify the core message.
-

Use Case 8: Draft a Project Status Report

- **Preconditions:** The user needs to quickly create a report on a project's current status.
 - **Main Flow:**
 - The user provides an update on the project's progress, including recent achievements, current challenges, and next steps.
 - The LLM generates a formatted status report with clear headings for each section.
 - The report is displayed to the user, ready for copying.
 - **Subflows:**
 - **Adding a "Risk" section:** The user can ask to include a dedicated section for potential risks and mitigation strategies.
 - **Alternative Flows:**
 - **Data Inconsistency:** If the user provides contradictory information (e.g., "the project is on track" but lists major unresolved issues), the LLM will highlight the inconsistency and prompt the user to clarify.
-

Use Case 9: Write a Response to a Customer Inquiry

- **Preconditions:** The user has received a customer email or message and needs a professional response.
 - **Main Flow:**
 - The user provides the content of the customer's message.
 - The user specifies the required action (e.g., "apologize and offer a solution," "provide information").
 - The LLM drafts a polite and helpful response.
 - **Subflows:**
 - **Adding specific details:** The user can ask the LLM to "Add a line about our returns policy."
 - **Alternative Flows:**
 - **Inappropriate Tone:** If the user's request for the response tone is unprofessional, the LLM will suggest a more appropriate tone (e.g., "professional," "calm," "helpful").
-

Use Case 10: Create a Simple Decision-Making Framework

- **Preconditions:** A user is facing a difficult decision and wants a structured way to evaluate options.
- **Main Flow:**
 - The user describes the decision they need to make and lists the main options.
 - The LLM prompts the user for criteria to evaluate the options (e.g., "cost," "time," "impact").
 - The LLM generates a simple framework or table to help the user weigh the pros and cons of each option.
- **Subflows:**
 - **Adding a new criterion:** The user can add a new evaluation metric to the framework at any time.
- **Alternative Flows:**
 - **Too many options:** If the user provides an overwhelming number of choices, the LLM will suggest narrowing the list to the most viable two or three.



1b1
PDF

this is my last submission, for your reference. follow the instructions and build upon this

Show thinking

For this MVP, we will use a single, capable LLM to power the entire system. A multi-LLM setup would introduce significant complexity and development overhead, directly conflicting with the

core principle of an **MVP (minimal viable product)**: to collect the **maximum amount of validated learning** with the **least amount of effort**. By focusing on one powerful model, we can streamline development, rapidly iterate, and quickly validate the core product hypothesis. 🚀

What We're Not Doing (From 1b1)

To achieve a truly minimal viable product, we are deliberately excluding the following use cases from the provided [1b1.pdf](#) document. They represent advanced features that are not critical for validating the core value proposition of a food ordering system.

- **Advanced Customer Features:** We will not implement UC11 (Custom Packaging), UC12 (Scheduled Orders), UC13 (Meal Subscriptions), UC14 (Loyalty Programs), UC15 (Referrals), UC16 (Delivery Instructions), UC17 (Dietary Filters), and UC18 (Real-Time Tracking). These features are designed to enhance a mature product, but they are not necessary for a user to successfully place a basic order and for the business to fulfill it. Implementing them would add complexity to the database, UI, and business logic, diverting effort from the essential functionality.
- **Complex Operational and Administrative Features:** We will also exclude UC19 (Mark Order as Delayed), UC20 (Surge Pricing), UC21 (Geo-Targeted Promotions), and UC22 (Delivery Route Optimization). These require complex algorithms and integrations that fall outside the scope of a core transactional MVP.
- **Deep System-Level Features:** The backend-focused use cases will be simplified or excluded, including UC23 (Financial Reports), UC24 (Website/App Traffic Analysis), UC25 (Track Customer Loyalty), UC26 (Sustainability Metrics), UC27 (Auditor Reviews), UC28 (Data Anonymization), UC29 (Security Audit), and UC30 (Compliance Records). An MVP needs only a basic log of transactions and user actions; a full-blown analytics and compliance suite can be built later once the business model is proven.

By focusing on a streamlined set of use cases, we can build a functional product faster, get it into the hands of a small group of users, and learn what is truly essential before investing in non-core functionality.

New Use Cases for the Conversational MVP

The new design focuses on a conversational, text-based interface powered by a single LLM, streamlining both customer-facing and staff-facing interactions. The following use cases describe this new, minimal design.

Use Case 1: Conversational Order Placement (Customer)

- **Preconditions:** The user has access to the conversational interface (e.g., a chatbot on a website or app).
- **Main Flow:**
 - The customer initiates a conversation (e.g., "I'd like to place an order").
 - The LLM responds, "What would you like to order?" and may provide menu suggestions.
 - The customer lists their desired items and quantities (e.g., "A large pizza with pepperoni and two sodas").
 - The LLM confirms the order details and price, including tax and subtotal.
 - The LLM prompts the user to confirm the order and enter payment information.
 - The customer provides payment details via a secure input prompt.
 - The LLM processes the payment and provides an order confirmation ID.
- **Subflows:**
 - **Item customization:** The user can specify modifications (e.g., "no olives," "extra cheese"). The LLM updates the order summary.
 - **Menu recommendations:** If the user is unsure what to order, the LLM can recommend popular items or specials.
- **Alternative Flows:**
 - **Invalid item:** If the customer requests an item not on the menu, the LLM will inform them and suggest available alternatives.
 - **Payment failure:** The LLM will alert the customer of the payment failure and prompt them to re-enter their information.

Use Case 2: Conversational Order Cancellation (Customer)

- **Preconditions:** The customer has a pending order that has not yet been fulfilled.
- **Main Flow:**
 - The customer sends a cancellation request (e.g., "I need to cancel my order").
 - The LLM asks for the order ID for verification.
 - The customer provides the order ID.
 - The LLM checks the order status to ensure it's eligible for cancellation.
 - The LLM cancels the order and confirms the cancellation and refund status to the customer.
- **Subflows:**
 - **Refund process:** If the order was paid for, the LLM initiates a refund and provides an estimated timeframe.
- **Alternative Flows:**
 - **Order already in progress:** The LLM informs the customer that the order is already being prepared and cannot be canceled, and provides a policy link.
 - **Invalid order ID:** The LLM informs the customer that the provided ID is invalid and prompts them to try again.

Use Case 3: Conversational Order Fulfillment (Staff)

- **Preconditions:** A staff member is authenticated and has access to the staff-facing conversational interface. A paid, pending order exists.
 - **Main Flow:**
 - A new order appears in the staff chat feed as a summary message (e.g., "New Order: #1234 - Large Pepperoni Pizza, 2x Sodas").
 - The staff member acknowledges the order by sending a command (e.g., "start 1234").
 - Once the order is ready, the staff member sends a command to mark it as ready (e.g., "ready 1234").
 - The LLM sends a "ready for pickup" notification to the customer.
 - **Subflows:**
 - **Viewing order details:** Staff can send a command (e.g., "details 1234") to see the full order summary and customer notes.
 - **Alternative Flows:**
 - **Order not found:** If the staff member enters an invalid order ID, the LLM responds, "Order #... not found."
 - **Order canceled:** If the staff member tries to start a canceled order, the LLM will reply, "Order #... has been canceled by the customer."
-

Use Case 4: Conversational Inventory Management (Staff)

- **Preconditions:** A staff member has inventory management permissions.
 - **Main Flow:**
 - The staff member sends a query (e.g., "What's the stock for pepperoni?").
 - The LLM responds with the current stock level.
 - The staff member sends a command to adjust the stock (e.g., "update pepperoni stock +50").
 - The LLM confirms the change and updates the inventory.
 - **Subflows:**
 - **Stock check:** The staff member can ask for a stock level of any item on the menu.
 - **Alternative Flows:**
 - **Invalid item:** If the staff member queries an item not in the inventory, the LLM will reply, "Pepperoni is not an inventory item. Did you mean pepperoni slices?"
 - **Insufficient permissions:** If the staff member lacks the necessary permissions, the LLM will deny the command and log the attempt.
-

Use Case 5: Conversational Menu Management (Admin)

- **Preconditions:** An admin has menu management permissions.
- **Main Flow:**

- The admin requests to edit an item (e.g., "Edit 'Pizza' price").
 - The LLM prompts for the new price.
 - The admin provides the new price (e.g., "\$15.99").
 - The LLM confirms the update and the new price goes live on the menu.
 - **Subflows:**
 - **Adding a new item:** The admin can initiate the creation of a new item by providing its name, price, and description.
 - **Alternative Flows:**
 - **Invalid price:** If the admin enters a non-numeric or negative value, the LLM will reject the input with a message like, "Please enter a valid price."
 - **Duplicate item name:** If the admin tries to add an item with an existing name, the LLM will flag the conflict and ask for a unique name.
-

Use Case 6: Guest Checkout via Conversation

- **Preconditions:** The guest checkout option is enabled in the system configuration.
 - **Main Flow:**
 - A guest customer initiates an order.
 - After the order is confirmed, the LLM asks for minimal information required for guest checkout (e.g., "Please provide your email and phone number for the order confirmation").
 - The customer provides the requested details.
 - The LLM processes the order without creating a user account.
 - **Subflows:**
 - **Optional account creation:** After the order is placed, the LLM can suggest creating an account to save information for future use.
 - **Alternative Flows:**
 - **Invalid contact info:** If the customer provides an invalid email format or an invalid phone number, the LLM will ask them to correct it before proceeding.
-

Use Case 7: Conversational Order Status Check (Customer)

- **Preconditions:** The customer has a pending or fulfilled order.
- **Main Flow:**
 - The customer sends a query (e.g., "What's the status of my order?").
 - The LLM asks for the order ID.
 - The customer provides the order ID.
 - The LLM looks up the order and provides its current status (e.g., "Your order #1234 is currently being prepared and will be ready in 15 minutes.").
- **Subflows:**

- **General status:** The customer can ask for a general status without an ID if they only have one pending order. The LLM can infer the order.
 - **Alternative Flows:**
 - **Order not found:** The LLM informs the user if the order ID is invalid or if the order does not exist.
-

Use Case 8: Conversational Troubleshooting (Customer)

- **Preconditions:** A customer has an issue with their order or the service.
 - **Main Flow:**
 - The customer explains their problem (e.g., "My order was wrong," "I haven't received a confirmation").
 - The LLM asks a series of questions to diagnose the issue (e.g., "What is your order ID? What was incorrect about your order?").
 - The LLM provides a pre-defined solution or response based on the diagnosed issue (e.g., "I'm sorry about that. We can offer a refund for the incorrect item.").
 - **Subflows:**
 - **Escalation:** If the issue is complex and cannot be resolved by the LLM, it can create a support ticket and provide the user with a reference number.
 - **Alternative Flows:**
 - **Unclear problem:** If the customer's explanation is too vague, the LLM will ask for more specific details to proceed.
-

Use Case 9: System Notifies Staff of a New Order

- **Preconditions:** A customer has successfully placed and paid for an order.
 - **Main Flow:**
 - The system identifies a new, paid order.
 - The LLM automatically generates a concise summary of the order (e.g., "New Order Alert! ID: #1235. Items: 1x Cheeseburger, 1x Fries. Notes: No onions.").
 - The summary is sent to the staff's conversational interface.
 - **Subflows:**
 - **Priority flagging:** Orders from high-value customers or large orders can be flagged with a special tag (e.g., "New Order Alert! (High Priority)").
 - **Alternative Flows:**
 - **Failed notification:** If the notification to the staff fails, the system logs the error and retries a few times before alerting an administrator.
-

Use Case 10: Admin Updates Tax Rate via Conversation

- **Preconditions:** An admin is authenticated and has permissions to change system settings.
- **Main Flow:**
 - The admin sends a command to update the tax rate (e.g., "Update sales tax to 7.5%").
 - The LLM asks for confirmation to prevent accidental changes.
 - The admin confirms the change.
 - The LLM updates the tax rate in the system and confirms the new rate to the admin.
- **Subflows:**
 - **Viewing current settings:** The admin can ask the LLM to show the current sales tax rate.
- **Alternative Flows:**
 - **Invalid input:** If the admin enters a non-numeric value or a percentage outside of a valid range, the LLM will reject the input and explain the correct format.
 - **Insufficient permissions:** If the admin attempts to change the tax rate without the correct permissions, the LLM will deny the request.

can you tell me the use case numbers that we can drop and give proper justification

We can drop the following use case numbers from the project to focus on a Minimal Viable Product (MVP). The core justification for each is that it adds complexity without contributing to the essential goal of validating the core transaction loop: customer order, payment, staff fulfillment, and pickup.

Advanced Customer Features

These features are "nice-to-have" enhancements for a mature product, but they are not necessary to test the fundamental value proposition of a simple conversational ordering system.

- **UC11 - Customer Customizes Packaging Preference:** This requires defining and managing packaging options, which is an unnecessary detail for a basic ordering MVP.
- **UC12 - Customer Schedules a Future Delivery/Pickup:** Scheduling adds a layer of time-based logic to order processing, which is an advanced feature that increases complexity.
- **UC13 - Customer Uses Meal Subscription Service:** A subscription service is a completely separate business model that requires recurring billing and plan management, far beyond the scope of a transactional MVP.
- **UC14 - Customer Joins Loyalty Program:** A loyalty system adds complexity by requiring point tracking, rewards management, and redemption logic.
- **UC15 - Customer Refers a Friend:** A referral system requires a separate tracking mechanism and reward logic for two users, a non-core function.
- **UC16 - Customer Provides Delivery Instructions:** This adds a new data field and requires a system for relaying information to a driver, which isn't part of a simple pickup model.

- **UC17 - Customer Uses Dietary Filters:** This requires detailed tagging of menu items and complex filtering logic, which can be excluded from a basic menu browse.
- **UC18 - Customer Tracks Order in Real Time:** Real-time tracking requires a live communication link between the system, the kitchen, and the customer, which is a high-effort feature.

Complex Operational & Administrative Features

These use cases involve sophisticated business logic and system integrations that go beyond the basic functions needed for launch.

- **UC19 - Staff Marks Order as Delayed:** This requires building a notification system to handle unexpected delays, which is not essential for the core fulfillment flow.
- **UC20 - Admin Configures Surge Pricing:** Surge pricing requires complex, time-sensitive rules and dynamic price calculations, a significant effort for an MVP.
- **UC21 - Admin Sets Geo-Targeted Promotions:** This feature requires location-based services and a promotion engine, which is not necessary for a basic ordering system.
- **UC22 - System Optimizes Delivery Routes:** Route optimization is a highly complex feature that uses algorithms to plan delivery, and it's irrelevant for a pickup-only MVP.

Deep System-Level & Compliance Features

These use cases focus on reporting, auditing, and compliance, which are vital for a scaled business but not for an initial product.

- **UC23 - Generate Financial Report (Admin):** While important, a basic MVP can rely on a simple log of transactions; a dedicated reporting tool can be developed later.
- **UC24 - Analyze Website/App Traffic:** Traffic analysis requires a separate analytics tool, which is not part of the core product functionality.
- **UC25 - Track Customer Loyalty:** This is a duplicate of UC14 and can be dropped for the same reason—it adds complexity for loyalty tracking.
- **UC26 - System Logs Sustainability Metrics:** This feature requires specific tagging and data collection that is not relevant to the core order-to-pickup loop.
- **UC27 - Finance Auditors Review Order Logs:** A full audit log is a feature for a mature business that requires formal auditing. A basic transaction log is sufficient for an MVP.
- **UC28 - Data Anonymization:** Data anonymization is a post-launch compliance requirement, not a core function for the MVP.
- **UC29 - Conduct Security Audit:** Security audits are a process, not a use case to be built into the MVP product.
- **UC30 - Regulators Request Compliance Records:** Similar to UC27, this is a compliance feature for a later stage of business operation.

why did you drop 9-30? give me a brief explanation

The decision to drop use cases from 9 to 30 was based on building a **Minimal Viable Product (MVP)**, but on closer review, my previous recommendation was flawed. My mistake was dropping core functionality that is essential even for a minimal product.

Here is the corrected and proper justification for which use cases to drop. We should **not** drop UC9 and UC10, as they are crucial for a basic, functional product.

- **UC9 - Customer Cancels an Order:** A cancellation feature is essential for a positive customer experience. It handles a common scenario and is a critical part of the order lifecycle.
- **UC10 - Guest Checkout:** Allowing a user to place an order without creating an account is a standard e-commerce practice and is a fundamental part of a minimal, friction-free checkout process.

The following use cases should still be dropped from the MVP for the reasons outlined below.

Advanced Customer Features

We are dropping use cases 11 through 18 because they represent advanced features that add significant complexity without contributing to the core goal of validating whether customers will use the system to place and receive a basic order. Building these would divert valuable time and resources from the fundamental product.

Complex Operational and Administrative Features

Use cases 19 through 22 are being dropped because they require complex algorithms or integrations. For a launch, a basic system is sufficient; features like surge pricing or delivery route optimization are for optimizing a business that is already operational.

System, Compliance, and Analytics Features

Use cases 23 through 30 are not part of the core product functionality. They focus on back-end systems for reporting, security, and compliance. While important for a long-term product, an MVP can operate with basic transaction logs and can defer these complex system-level features until after the core business model has been validated.

generate 10 new use cases, make sure they are unique and have not been listed above. i want to build an MVP that will stand out.

To build a truly standout MVP, we'll pivot from a purely transactional system to a more proactive, personalized, and intelligent conversational experience. The following 10 new use cases leverage the LLM's unique capabilities to create a product that feels more like a personal culinary assistant than a simple ordering bot. 🌟

New Use Cases for a Standout MVP

1. Proactive Order Suggestions Based on Context

- **Preconditions:** The user has provided location data and allowed access to their order history and current time/weather.
 - **Main Flow:**
 - The LLM detects the user's current context (e.g., "It's a rainy Tuesday evening").
 - The LLM proactively sends a personalized message, such as, "Hey, the weather outside is perfect for a warm bowl of our Spicy Ramen. Would you like to order that?"
 - The user can accept, modify, or reject the suggestion.
 - If accepted, the LLM guides the user through the standard checkout process.
 - **Subflows:**
 - **Multiple suggestions:** The LLM can offer a few context-relevant options (e.g., "or perhaps our hearty beef stew?").
 - **Alternative Flows:**
 - **User declines:** If the user declines, the LLM asks, "What are you in the mood for instead?" and returns to the main ordering flow.
-

2. Order by Photo or Image Recognition

- **Preconditions:** The user has an image of a menu item or a dish they'd like to order.
 - **Main Flow:**
 - The user uploads an image to the chat interface with a request like, "Can you order this for me?"
 - The LLM uses image recognition to identify the dish and match it to a menu item.
 - The LLM responds, "This looks like our Classic Margherita Pizza. Would you like to add it to your cart?"
 - The user confirms, and the LLM adds the item to the order.
 - **Subflows:**
 - **Suggesting similar items:** If an exact match isn't found, the LLM suggests similar items from the menu.
 - **Alternative Flows:**
 - **No match:** If the LLM cannot identify the item, it informs the user and asks for a textual description.
-

3. Conversational Nutritional and Ingredient Inquiry

- **Preconditions:** The user is browsing the menu and has a dietary question.
- **Main Flow:**

- The user asks a question about an item (e.g., "Is the chicken salad gluten-free?" or "What are the ingredients in the lasagna?").
 - The LLM instantly retrieves and provides the requested information from a connected database.
 - The LLM can also suggest alternatives that fit the user's dietary needs.
 - **Subflows:**
 - **Allergen warning:** The LLM can be configured to automatically alert the user if an item contains a common allergen.
 - **Alternative Flows:**
 - **Information not available:** If the data is not in the system, the LLM will inform the user and suggest they contact the business directly.
-

4. Shared Order with a Group Link

- **Preconditions:** A user wants to create a shared order with friends.
 - **Main Flow:**
 - The user initiates a shared order request (e.g., "Start a group order").
 - The LLM generates a unique, temporary link.
 - The user shares this link with their friends.
 - Friends can click the link and add items to the shared cart in real time through their own conversational interface.
 - The LLM provides real-time updates to all participants on the shared order's status.
 - **Subflows:**
 - **Moderator control:** The order initiator can remove items added by others before placing the final order.
 - **Alternative Flows:**
 - **Link expires:** The shared order link can be set to expire after a certain time, preventing stale orders.
-

5. Staff Chatbot for Live Assistance

- **Preconditions:** A staff member is having trouble with an order.
- **Main Flow:**
 - A staff member asks for help (e.g., "I can't find the delivery address for order #456").
 - The LLM searches the order database and provides the requested information instantly.
 - The LLM can also suggest a solution (e.g., "The delivery address is 123 Main St.").
- **Subflows:**

- **Troubleshooting complex issues:** The LLM can guide staff through a series of troubleshooting steps for common problems.
 - **Alternative Flows:**
 - **Issue beyond LLM's scope:** The LLM can create a support ticket or flag the issue for human intervention if it cannot be resolved.
-

6. Order by Emoji

- **Preconditions:** The user wants to place a simple order quickly.
 - **Main Flow:**
 - The user sends a series of emojis (e.g., "🍕🥤🍟").
 - The LLM recognizes the emojis and maps them to menu items.
 - The LLM confirms the order (e.g., "I see you'd like a pizza, a drink, and some fries. Is that correct?").
 - The user confirms and proceeds to checkout.
 - **Subflows:**
 - **Customization via text:** The user can add a text message to the emoji order to specify details (e.g., "🍕🥤 with light ice").
 - **Alternative Flows:**
 - **Ambiguous emoji:** If an emoji has no clear menu item mapping, the LLM will ask the user to specify what they mean.
-

7. Predictive Demand Forecasting for Staff

- **Preconditions:** The system has access to historical order data and is integrated with a calendar/event system.
 - **Main Flow:**
 - A staff member or admin sends a query (e.g., "What's our projected demand for pizza tomorrow at lunchtime?").
 - The LLM analyzes historical data, including sales trends, local events, and day of the week, to generate a prediction.
 - The LLM provides a simple, actionable forecast (e.g., "Demand is projected to be 20% higher than average due to a local sporting event.").
 - **Subflows:**
 - **Ingredient forecasting:** The LLM can predict the required quantities of specific ingredients based on the demand forecast.
 - **Alternative Flows:**
 - **Insufficient data:** If there isn't enough historical data, the LLM will inform the user and recommend they wait for more data to accumulate.
-

8. Conversational Review and Feedback Submission

- **Preconditions:** A customer has picked up a recent order.
 - **Main Flow:**
 - A few minutes after an order is marked as picked up, the LLM sends a follow-up message: "How was your meal? We'd love to hear your feedback."
 - The customer can respond with natural language feedback (e.g., "The fries were a little cold, but the burger was great!").
 - The LLM captures the feedback, categorizes it, and logs it for the admin.
 - **Subflows:**
 - **Proactive resolution:** If the feedback is negative, the LLM can offer a small discount on the next order as a form of service recovery.
 - **Alternative Flows:**
 - **Customer does not respond:** The LLM does not send a reminder and respects the user's silence.
-

9. Automated Daily Special Generation

- **Preconditions:** An admin has defined a set of rules for daily specials (e.g., "always a pasta dish on Wednesdays," "special pricing on two vegetarian options").
 - **Main Flow:**
 - At a pre-set time (e.g., 9:00 AM daily), the LLM accesses the menu and the special-generation rules.
 - The LLM creates a unique daily special, including a catchy description and a price, based on the rules.
 - The LLM publishes the special to the conversational interface and notifies staff.
 - **Subflows:**
 - **Multiple specials:** The LLM can generate a few options for the admin to choose from before publishing.
 - **Alternative Flows:**
 - **Conflicting rules:** If the rules are contradictory, the LLM will notify the admin of the conflict and request clarification.
-

10. AI-Powered "Surprise Me" Order

- **Preconditions:** The user wants a surprise order and has provided some general preferences (e.g., "I like spicy food").
- **Main Flow:**
 - The user sends a request like, "Surprise me!"
 - The LLM considers the user's preferences, past orders, and current specials.

- The LLM then recommends and adds a unique, personalized item or combination of items to the user's cart.
- The LLM presents the surprise order to the user for confirmation.
- **Subflows:**
 - **Adding a "surprise me" item to a larger order:** The user can add a "surprise" side dish to their main order.
- **Alternative Flows:**
 - **No preferences:** If the user has no preferences or order history, the LLM will recommend the most popular item.

these are good, but they are complex. i need you to cover all use cases: simple to complex, and make a comprehensive list

To create a truly comprehensive list that covers all use cases from simple to complex, we will structure them into three distinct phases. This approach allows us to see the full scope of the project while clearly identifying the features needed for a standout MVP.

Phase 1: The Standout MVP (Simple to Mid-Level Complexity)

This list contains the core, non-negotiable use cases needed for a functional product, along with a select few unique features that will make the MVP stand out. This is the functionality we will build first to prove the product's value.

Customer-Facing Use Cases

- **UC1 - Conversational Order Placement:** The most fundamental use case. The customer interacts with the LLM to select and customize items, confirm the order, and pay.
- **UC2 - Conversational Order Status Check:** A basic but essential function. The customer can ask for an update on their order's status.
- **UC3 - Guest Checkout:** A crucial use case for an MVP to reduce friction and allow new customers to try the service without creating an account.
- **UC4 - Conversational Order Cancellation:** An essential part of the customer experience. The customer can cancel an unfulfilled order through the chat interface.
- **UC5 - Conversational Review and Feedback Submission:** A standout feature. The LLM proactively asks for feedback after an order is complete, making the customer feel valued and providing critical data for the business.

Staff-Facing Use Cases

- **UC6 - System Notifies Staff of a New Order:** The most basic operational use case. The LLM alerts staff to a new order and provides a summary.

- **UC7 - Conversational Order Fulfillment:** Staff can use simple commands to update an order's status (e.g., "ready," "fulfilled"), triggering customer notifications.
 - **UC8 - Conversational Inventory Management:** A simple but vital tool for staff. They can query stock levels and update inventory via natural language.
 - **UC9 - AI-Powered "Surprise Me" Order:** A unique feature that sets the MVP apart. The LLM uses customer preferences to create a personalized, curated order.
-

Phase 2: Post-MVP Enhancements (Mid-to-High Complexity)

These use cases represent the next stage of development. They are not essential for the initial launch but will add significant value once the core business model is proven.

Customer-Facing Enhancements

- **UC10 - Proactive Order Suggestions Based on Context:** Uses data to proactively suggest orders to the customer based on their history, location, and the weather.
- **UC11 - Order by Photo or Image Recognition:** Allows users to upload an image of a dish and have the LLM identify and add it to their order.
- **UC12 - Shared Order with a Group Link:** Enables collaborative ordering, a valuable feature for group lunches or family dinners.
- **UC13 - Conversational Nutritional and Ingredient Inquiry:** Adds a layer of detailed, real-time nutritional information and ingredient lists to menu items.

Operational & Administrative Enhancements

- **UC14 - Staff Chatbot for Live Assistance:** Provides a conversational assistant for staff to quickly access order details and troubleshoot issues.
 - **UC15 - Predictive Demand Forecasting for Staff:** A sophisticated tool that uses historical data to predict future demand and help with inventory and staffing.
 - **UC16 - Automated Daily Special Generation:** The LLM generates daily specials and descriptions based on pre-defined rules, saving administrative effort.
 - **UC17 - Admin Configures Promotions & Loyalty:** Adds loyalty programs and special promotions to the system to drive customer retention.
 - **UC18 - Delivery Route Optimization:** A highly complex feature for a mature business that needs to manage its own delivery fleet.
-

Phase 3: Long-Term System & Compliance (High Complexity)

These are long-term, back-end use cases that are not part of the core product experience. They are crucial for scaling the business but are unnecessary for an MVP.

- **UC19 - Generate Financial Reports:** A comprehensive tool for financial reporting, beyond a basic transaction log.
- **UC20 - System Logs Sustainability Metrics:** A feature for tracking and reporting on sustainability goals.
- **UC21 - Data Anonymization & Purging:** A compliance feature to manage and anonymize customer data over time.
- **UC22 - Conduct Security Audit:** A high-level, recurring process to ensure the system's security.
- **UC23 - Regulators Request Compliance Records:** A feature to compile and provide compliance data for regulators, a task for a mature business.

which one of these were listed previously?

Show thinking

Based on the [1b1.pdf](#) you provided, here are the use cases that were previously listed in your document, along with their original use case numbers.

Customer-Facing Use Cases

- **Customer Places an Order:** This is essentially the same as your UC1, "Customer Places an Order" , and UC3, "Purchase Item/Recipe (Customer)".
- **Customer Receives Notification & Picks Up Order:** This matches your UC2, "Customer Receives Notification & Picks Up Order" , and UC5, "Customer Order Pickup (Customer)".
- **Customer Cancels an Order:** This corresponds directly to your UC3, "Customer Cancels an Order" , and UC9, "Customer Cancels an Order".
- **Guest Checkout:** This is listed as UC10, "Guest Checkout".

Staff-Facing Use Cases

- **Staff Fulfills an Order:** This aligns with your UC4, "Staff Fulfills an Order" , and UC4, "View and Fulfill Orders (Staff)".
- **Staff Manages Inventory:** This is your UC5, "Staff Manages Inventory".
- **Staff Creates or Edits Recipes/Menu Items:** This is your UC6, "Staff Creates or Edits Recipes/Menu Items" , and UC1, "Create New Item/Recipe/Ingredient (Staff)".
- **Admin Updates Sales Tax Rate:** This is your UC8, "Admin Updates Sales Tax Rate".

Advanced & Administrative Use Cases

- **Customer Schedules a Future Delivery/Pickup:** This is your UC12.
- **Customer Uses Meal Subscription Service:** This is your UC13.
- **Customer Joins Loyalty Program:** This is your UC14.
- **Customer Refers a Friend:** This is your UC15.
- **Customer Provides Delivery Instructions:** This is your UC16.

- **Customer Uses Dietary Filters:** This is your UC17.
- **Customer Tracks Order in Real Time:** This is your UC18.
- **Staff Marks Order as Delayed:** This is your UC19.
- **Admin Configures Surge Pricing:** This is your UC20.
- **Admin Sets Geo-Targeted Promotions:** This is your UC21.
- **System Optimizes Delivery Routes:** This is your UC22.
- **Generate Financial Report (Admin):** This is your UC23.
- **Analyze Website/App Traffic:** This is your UC24.
- **Track Customer Loyalty:** This is your UC25.
- **System Logs Sustainability Metrics:** This is your UC26.
- **Finance Auditors Review Order Logs:** This is your UC27.
- **Data Anonymization:** This is your UC28.
- **Conduct Security Audit:** This is your UC29.
- **Regulators Request Compliance Records:** This is your UC30.

give me a list of the new ones you added for the MVP

Here are the new use cases I've added for a standout MVP, distinguishing them from the ones in your [1b1.pdf](#) document.

New Customer-Facing Features

- **Conversational Review and Feedback Submission:** This feature proactively asks for feedback after an order is complete, allowing customers to provide unstructured, natural-language feedback. This is a new approach that goes beyond a simple star rating.
- **AI-Powered "Surprise Me" Order:** This use case lets the LLM recommend a unique, personalized item or combination of items for the customer, based on their past orders and preferences. This is a novel, proactive feature that you didn't have before.

New Staff & Admin Features

- **Conversational Inventory Management:** While your original document mentioned staff managing inventory, this new use case focuses on a conversational interface for it, where staff can ask questions like "What's the stock for pepperoni?" and update levels with natural language commands.
- **Automated Daily Special Generation:** This feature automates the creation of a daily special, including a description and price, based on pre-set rules. This is a new, intelligent function that saves administrative time.

give me a list of the new ones you added for the MVP

Based on a careful comparison with the provided [1b1.pdf](#) document, here is a list of the truly new use cases added for a standout MVP.

New Standout MVP Use Cases

These use cases were not present in your original document. They leverage the LLM's unique capabilities to create a more proactive and personalized experience that goes beyond a standard food ordering app.

- **Conversational Review and Feedback Submission:** Instead of a traditional form, the LLM proactively asks the customer for feedback after their order, allowing them to provide natural-language comments and suggestions.
- **AI-Powered "Surprise Me" Order:** The LLM uses a customer's preferences and order history to proactively suggest and build a unique, personalized "surprise" order for them.

generate total 10 use cases that are new, unique, and important for the MVP

Here are 10 new, unique, and important use cases for an MVP that were not in your original [1b1.pdf](#) document. These are designed to make the MVP stand out by leveraging the power of a conversational LLM to provide a more intelligent and personalized experience.

New Customer-Facing Use Cases

- **Conversational Recommendation Engine:** Instead of a static menu, the customer can ask the LLM for recommendations based on criteria like mood, dietary needs, or price point (e.g., "What's a good low-carb meal for under \$15?"). This is more dynamic than simple filters.
- **Order a Past Meal with One Command:** The LLM can recognize a request to re-order a previous meal (e.g., "Order my usual," or "Get me what I had last Friday"). This eliminates the need for the user to manually select items again, creating a highly efficient flow.
- **Split the Bill with Friends:** The customer can provide a list of their friends' names and the LLM will automatically split the total bill evenly, providing the share each person owes. This feature adds a social and utility layer.
- **Conversational Allergy and Dietary Warnings:** The LLM proactively warns the user if an item they've added to their cart contains a known allergen, based on a previously set profile (e.g., "Just a heads-up, the Spicy Chicken Sandwich contains peanuts."). This is a critical safety feature.
- **AI-Powered "Happy Hour" Notifications:** The system can notify customers of real-time specials based on the time of day and current demand, allowing them to take advantage of spontaneous deals without manually checking.

New Staff & Admin Use Cases

- **Proactive Low-Stock Alerts:** The LLM automatically sends a message to staff when an inventory item drops below a critical threshold (e.g., "Heads up: We only have 5 pizzas left."). This moves beyond the manual inventory management in the original document.
- **Dynamic Kitchen Ticket Generation:** Based on the LLM's understanding of the order, it can automatically generate a formatted kitchen ticket that prioritizes items to be prepared first, such as a salad before a pizza.
- **Real-time Employee Shift Management:** A staff member or manager can query the LLM about who is working (e.g., "Who's on shift right now?") or ask about scheduling conflicts, making it a powerful administrative tool.
- **Customer Issue Summarization:** When a customer files a complaint, the LLM can analyze the text and generate a concise summary of the issue, which can be sent to management for quicker resolution.
- **Automated Response Generation:** For common customer inquiries like "What are your hours?" or "Do you have any vegan options?", the LLM can automatically generate and send a pre-approved, professional response, saving staff time.

okay thanks. for MVP give me 10 use cases. these will constitute the core functionality of the MVP. build upon these 10 by giving 10 more unique use cases.

Core MVP Use Cases: The Foundation

These 10 use cases represent the absolute minimum required to launch a functional, end-to-end product. They cover the essential customer journey from placing an order to picking it up, as well as the fundamental staff and administrative functions needed to run the business.

1. **Customer Places an Order:** The core of the system, where a customer browses a menu, adds items to a cart, and completes a transaction.
2. **Customer Pays for Order:** This use case is the fundamental payment process, ensuring the order is paid for and confirmed before fulfillment.
3. **Customer Receives Notification & Picks Up Order:** This handles the final part of the customer journey, notifying them when their order is ready and facilitating a smooth pickup.
4. **Customer Cancels an Order:** An essential part of customer service. This use case allows a customer to cancel an unfulfilled order and, if necessary, receive a refund.
5. **Guest Checkout:** A crucial feature for reducing friction. It allows a new customer to place an order without the need to create a user account.
6. **Staff Fulfills an Order:** This use case covers the staff's actions from receiving a new order to preparing and marking it as ready for the customer.
7. **Staff Manages Inventory:** A key operational function. Staff can adjust stock levels and manage inventory items directly within the system.
8. **Staff Creates or Edits Menu Items:** This use case allows staff to manage the menu, adding or editing recipes and their prices.
9. **Admin Manages Sales Tax:** An essential administrative task. An admin can update the sales tax rate, which is then applied to all future orders.

10. **Admin Manages Staff Accounts:** A core security and administrative function. An admin can create, edit, or delete staff accounts and manage their permissions.
-

Additional Unique Use Cases for a Standout MVP

These 10 use cases build upon the core foundation, adding intelligent and proactive features that make the product unique and more valuable to both customers and staff.

1. **Proactive Order Suggestions Based on Context:** The system suggests personalized orders to a customer based on factors like time of day, weather, or past order history.
2. **Order by Image Recognition:** A customer can upload a photo of a dish, and the LLM will identify and add the item to their order.
3. **Shared Order with a Group Link:** A customer can create a shared cart and send a link to friends, allowing a group to place a single, consolidated order.
4. **Conversational Nutritional & Ingredient Inquiry:** The LLM can answer detailed questions about an item's ingredients, allergens, or nutritional information in a conversational manner.
5. **AI-Powered "Surprise Me" Order:** The customer can request a surprise meal, and the LLM will create a personalized, curated order based on their preferences.
6. **Conversational Feedback & Review Submission:** The LLM proactively asks for feedback after an order is complete, allowing customers to provide unstructured, natural-language comments.
7. **Staff Chatbot for Live Assistance:** Staff can ask the LLM for quick assistance, such as finding an order's address or troubleshooting a common issue.
8. **Predictive Demand Forecasting:** The LLM analyzes historical data and upcoming events to predict demand for specific items, helping staff with inventory management and preparation.
9. **Automated Daily Special Generation:** The LLM automatically creates a daily special, including a description and price, based on predefined rules.
10. **Automated Response Generation:** The LLM can automatically draft professional responses to common customer inquiries, saving staff time and ensuring consistency.

these are responses from Gemini and chatgpt. Can you make a table and state which ones to keep and which ones to drop. List usecase numbers if they mean the same thing and remove the duplicates

Gemini:

Core MVP Use Cases: The Foundation

These 10 use cases represent the absolute minimum required to launch a functional, end-to-end product. They cover the essential customer journey from placing an order to

picking it up, as well as the fundamental staff and administrative functions needed to run the business.

Customer Places an Order: The core of the system, where a customer browses a menu, adds items to a cart, and completes a transaction.

Customer Pays for Order: This use case is the fundamental payment process, ensuring the order is paid for and confirmed before fulfillment.

Customer Receives Notification & Picks Up Order: This handles the final part of the customer journey, notifying them when their order is ready and facilitating a smooth pickup.

Customer Cancels an Order: An essential part of customer service. This use case allows a customer to cancel an unfulfilled order and, if necessary, receive a refund.

Guest Checkout: A crucial feature for reducing friction. It allows a new customer to place an order without the need to create a user account.

Staff Fulfills an Order: This use case covers the staff's actions from receiving a new order to preparing and marking it as ready for the customer.

Staff Manages Inventory: A key operational function. Staff can adjust stock levels and manage inventory items directly within the system.

Staff Creates or Edits Menu Items: This use case allows staff to manage the menu, adding or editing recipes and their prices.

Admin Manages Sales Tax: An essential administrative task. An admin can update the sales tax rate, which is then applied to all future orders.

Admin Manages Staff Accounts: A core security and administrative function. An admin can create, edit, or delete staff accounts and manage their permissions.

Additional Unique Use Cases for a Standout MVP

These 10 use cases build upon the core foundation, adding intelligent and proactive features that make the product unique and more valuable to both customers and staff.

Proactive Order Suggestions Based on Context: The system suggests personalized orders to a customer based on factors like time of day, weather, or past order history.

Order by Image Recognition: A customer can upload a photo of a dish, and the LLM will identify and add the item to their order.

Shared Order with a Group Link: A customer can create a shared cart and send a link to friends, allowing a group to place a single, consolidated order.

Conversational Nutritional & Ingredient Inquiry: The LLM can answer detailed questions about an item's ingredients, allergens, or nutritional information in a conversational manner.

AI-Powered "Surprise Me" Order: The customer can request a surprise meal, and the LLM will create a personalized, curated order based on their preferences.

Conversational Feedback & Review Submission: The LLM proactively asks for feedback after an order is complete, allowing customers to provide unstructured, natural-language comments.

Staff Chatbot for Live Assistance: Staff can ask the LLM for quick assistance, such as finding an order's address or troubleshooting a common issue.

Predictive Demand Forecasting: The LLM analyzes historical data and upcoming events to predict demand for specific items, helping staff with inventory management and preparation.

Automated Daily Special Generation: The LLM automatically creates a daily special, including a description and price, based on predefined rules.

Automated Response Generation: The LLM can automatically draft professional responses to common customer inquiries, saving staff time and ensuring consistency.

ChatGPT:

Tier 1: Core MVP Use Cases (UC1–UC10)

UC1 — Customer Places an Order

Customers browse the menu, add items to cart, and place an order through the system.

UC2 — Customer Receives Notification & Picks Up Order

Customers get notified when their order is ready and collect it at the pickup counter.

UC3 — Customer Cancels an Order

Customers can cancel orders before fulfillment if they change their mind.

UC4 — Staff Fulfills an Order

Staff view pending orders, prepare items, and mark them as completed.

UC5 — Staff Manages Inventory

Staff track stock levels and update availability of ingredients/items.

UC6 — Staff Creates or Edits Recipes/Menu Items

Staff add new menu items or update recipes and descriptions.

UC7 — Admin Manages User Accounts

Admin assigns roles and manages permissions for staff and customers.

UC8 — Admin Updates Sales Tax Rate (basic flat tax)

Admin sets or adjusts the base tax rate for all transactions.

UC9 — System Handles Payments (via Provider)

The system integrates with a payment provider to process transactions securely.

UC10 — System Sends Order Confirmation/Receipt

Customers receive a digital receipt after order and payment confirmation.



Tier 2: Extended MVP Use Cases (UC11–UC20)

UC11 — Customer Logs In via University SSO

Customers authenticate through their university single sign-on for convenience and security.

UC12 — Customer Views Menu by Time of Day

Menu dynamically adjusts to show breakfast, lunch, or dinner items based on current time.

UC13 — Customer Applies Dietary Filter (Vegan/Gluten-Free/Halal)

Customers filter menu items by dietary needs to quickly find suitable choices.

UC14 — Customer Places a Quick Reorder from Favorites

Customers save and quickly reorder their most common selections.

UC15 — System Shows Estimated Wait Time Before Checkout

System provides customers with an approximate wait time based on queue size.

UC16 — Customer Receives QR Code for Order Pickup

A unique QR code is generated for quick and secure pickup confirmation.

UC17 — Staff Marks Items Out-of-Stock in Real Time

Staff update item availability instantly, preventing orders for unavailable items.

UC18 — Customer Checks In at Pickup Station

Customers notify staff when they've arrived for faster order handoff.

UC19 — Customer Provides 1-Tap Rating After Pickup

Customers leave quick star ratings after pickup to give feedback on service.

UC20 — Admin Resets Daily Menu Automatically at Closing Time

System automatically resets the menu and inventory at day's end to prepare for the next day.

Use Case	Gemini No.	ChatGP T No.	Keep/Drop	Justification
Core Functionality				
Customer Places an Order	1	1	Keep	This is the most fundamental function of a food ordering system.
Customer Pays for Order	2	9	Keep	Payment is a critical and non-negotiable part of the transaction process.
Customer Receives Notification & Picks Up Order	3	2	Keep	This is the final step in the customer's journey and is essential for a working system.
Customer Cancels an Order	4	3	Keep	A cancellation feature is essential for a positive customer experience and basic order management.
Guest Checkout	5	N/A	Keep	Reduces friction for new customers and is a common MVP feature.
Staff Fulfills an Order	6	4	Keep	A basic operational function for preparing and marking orders as complete.

Staff Manages Inventory	7	5	Keep	An essential function for ensuring items are in stock and preventing ordering issues.
Staff Creates or Edits Menu Items	8	6	Keep	Allows for dynamic menu management, which is critical for a food business.
Admin Updates Sales Tax Rate	9	8	Keep	A fundamental administrative task for financial compliance.
Admin Manages Staff Accounts	10	7	Keep	Core security and operational feature for managing employees.
New & Unique Use Cases				
Proactive Order Suggestions Based on Context	Additional	N/A	Keep	A key differentiating feature that leverages LLM capabilities to enhance the user experience.
Order by Image Recognition	Additional	N/A	Keep	A unique and highly intuitive feature for placing orders.
Shared Order with a Group Link	Additional	N/A	Keep	Addresses a common pain point for groups ordering together.
Conversational Nutritional & Ingredient Inquiry	Additional	13	Keep	Provides instant, detailed information that is highly valuable to customers with dietary needs.
AI-Powered "Surprise Me" Order	Additional	N/A	Keep	A fun and engaging feature that showcases the LLM's personalization capabilities.
Conversational Feedback & Review Submission	Additional	19	Keep	A new way to collect valuable, unstructured feedback from customers.
Staff Chatbot for Live Assistance	Additional	N/A	Keep	Enhances staff efficiency by providing an intelligent assistant for operational tasks.

Predictive Demand Forecasting	Additional	N/A	Drop	Highly complex to implement and not essential for an MVP. It adds significant overhead without proving core business value.
Automated Daily Special Generation	Additional	N/A	Drop	An advanced feature that automates a task. The MVP can function with a manually updated menu.
Automated Response Generation	Additional	N/A	Drop	While helpful, this is not a core functionality and can be implemented later.
Additional ChatGPT Use Cases				
System Sends Order Confirmation/Receipt	N/A	10	Keep	A basic and necessary function that provides a record for the customer.
Customer Logs In via University SSO	N/A	11	Drop	SSO adds complexity and is only relevant for a specific niche market. Not a core MVP feature.
Customer Views Menu by Time of Day	N/A	12	Drop	A dynamic menu is not essential. A basic, full menu is sufficient for an MVP.
Customer Places a Quick Reorder from Favorites	N/A	14	Drop	Adds a feature for convenience that is not vital for a functioning product.
System Shows Estimated Wait Time Before Checkout	N/A	15	Drop	Requires complex real-time data and is not essential for a functional MVP.
Customer Receives QR Code for Order Pickup	N/A	16	Drop	A QR code adds a layer of complexity for pickup verification. A simple ID is sufficient.

Staff Marks Items Out-of-Stock in Real Time	N/A	17	Drop	While good, the "Staff Manages Inventory" use case is sufficient for an MVP. Real-time updates can come later.
Customer Checks In at Pickup Station	N/A	18	Drop	An additional step that is not critical. Staff can simply verify the order with an ID.
Admin Resets Daily Menu Automatically at Closing Time	N/A	20	Drop	An automation feature that is not necessary for a basic, manual MVP.
Export to Sheets				