SOFTWARE ENGINEERING
PROJECT 1b1

30 Use Cases

UC-1. Peak Hour Ordering with Promotions

Preconditions: Customer has an active account; peak hour promotions active; customer in delivery zone.

Main Flow:

- Customer opens app during peak hours.
- System displays targeted promotions (e.g., for female users).
- Customer adds 2–4 items to cart to qualify.
- System applies automatic discounts.
- Customer places order; acknowledges extended delivery.
- System queues order with peak hour priority.

  Alternative Flows:
- Promotion expired → system offers next promotion.
- Delivery overload → system suggests pre-ordering.

UC-2. Group Social Ordering Experience

Preconditions: Multiple customers with accounts; group ordering enabled; shared delivery location confirmed.

Main Flow:

- Primary customer initiates group order.
- Invitations sent via app/social media.
- Each participant adds items to shared cart.
- System manages dietary restrictions.
- Payment split or managed by primary customer.
- Single delivery coordinated.

  Alternative Flows:
- Participant drops out → recalc order and fees.
- Location mismatch → alternative meeting point suggested.

UC-3. Health-Conscious Food Selection

Preconditions: Customer has dietary preferences; nutritional database integrated.

Main Flow:

- Customer applies health filters (low-calorie, gluten-free).

- System highlights balanced options.
- Customer views detailed nutritional info.
- System suggests healthier alternatives.
- Order placed with health tracking integration.

Alternative Flows:

- Limited healthy options → suggest better restaurants.
- Nutritional data unavailable → provide disclaimers.

UC-4. Reorder a Past Meal

Preconditions: Customer logged in with past order history.

Main Flow:

- Customer selects previous order from history.
- System adds items to current cart.
- Customer confirms/modifies cart; proceeds to checkout.

Alternative Flows:

- Item unavailable → system notifies for replacement/removal.

UC-5. Report an Issue with an Order

Preconditions: Customer received order.

Main Flow:

- Customer selects “Report an Issue.”
- Chooses reason (missing item, cold food, etc.) and description.
- Submits report → system notifies restaurant/admin.

Alternative Flows:

- Issue not valid for refund → system explains policy, offers credit.

UC-6. Small Restaurant Digital Onboarding

Preconditions: Owner has registration docs; digital infrastructure; onboarding team assigned.

Main Flow:

- Owner initiates registration.
- Platform provides step-by-step wizard.
- Menu digitized; photos uploaded.
- POS integrated; payment and analytics setup.
- Restaurant goes live with promotions.

Alternative Flows:

- Technical difficulties → human support assists.
- POS incompatibility → alternative solutions provided.

UC-7. Menu Performance Optimization
Preconditions: Restaurant operating ≥30 days; order analytics available.
Main Flow:

- Owner accesses analytics dashboard.
- System displays performance metrics and feedback.
- Platform suggests optimizations.
- Owner implements changes; system tracks improvements.
  Alternative Flows:
- Performance decline → system provides recovery strategy.
- Limited data → use industry benchmarks.

UC-8. Real-Time Kitchen Order Management
Preconditions: Digital kitchen display installed; staff trained.
Main Flow:

- Orders appear on kitchen screens.
- Staff mark prep/cooking/ready stages.
- System updates customers and drivers.
  Alternative Flows:
- Display failure → fallback to printed tickets.
- Staff override → manager adjusts priorities.

UC-9. Manage Menu Availability
Preconditions: Staff logged in.
Main Flow:

- Staff updates item availability.
- System updates customer-facing menu.
  Alternative Flows:
- Trying to mark “In Stock” without inventory → warning displayed.

UC-10. Manage Inventory Alerts
Preconditions: Staff logged in.
Main Flow:

- Staff sets low-inventory thresholds.
- System notifies staff when thresholds reached.

- Staff reorders ingredients.

Alternative Flows:

- Ingredient reaches zero → related items marked “Sold Out.”

UC-11. Driver Flexible Scheduling System
Preconditions: Driver onboarded; scheduling system shows shifts.
Main Flow:

- Driver submits preferred schedule.
- System assigns shifts considering demand and preferences.
- Driver can drop shifts; open shifts available for others.

Alternative Flows:

- No available shifts → waitlist provided.
- Schedule conflict → request adjustments with dispatcher.

UC-12. Driver Route Optimization with Human Override
Preconditions: Active delivery assignments; GPS/traffic operational.
Main Flow:

- System provides route optimization.
- Driver can override route due to road/safety.
- Dispatcher confirms changes if needed.

Alternative Flows:

- Safety concerns → driver refuses delivery.
- Customer location issues → alternative coordination.

UC-13. Driver Earnings Transparency
Preconditions: Verified driver; payment system integrated.
Main Flow:

- Driver views delivery pay breakdown.
- Platform ensures minimum wage guarantee.
- Reports include tips, incentives.

Alternative Flows:

- Payment delays → backup processing activated.
- Earnings dispute → system provides detailed history.

UC-14. Driver Communication and Human Support
Preconditions: Active delivery assignment; dispatchers available.
Main Flow:

- Driver contacts dispatcher for issues.
- Dispatcher provides guidance; system logs resolution.
  Alternative Flows:
- Dispatcher unavailable → automated guidance with callback.
- Emergency situations → emergency protocols activated.

UC-15. Accept or Reject an Order
Preconditions: Driver logged in and available.
Main Flow:

- Driver reviews order; selects Accept.
- System updates order status; notifies customer/restaurant.
  Alternative Flows:
- Reject → system sends order to next driver.
- No response → system auto-assigns to next driver.

UC-16. Crisis Management
Preconditions: Admin has full system access; monitoring systems active.
Main Flow:

- Automated systems detect potential disruptions.
- Admin receives alerts, implements immediate responses.
- Coordinates with stakeholders to minimize impact.
  Alternative Flows:
- System-wide failure → emergency manual operation.
- External dependencies → coordinate with third-party providers.

UC-17. Proactive System Monitoring and Issue Detection
Preconditions: Monitoring systems deployed; predictive models trained.
Main Flow:

- System monitors KPIs.
- Predictive models identify issues early.
- Admin takes preventive action; system learns from outcomes.
  Alternative Flows:
- False positive alerts → fine-tune detection algorithms.

- Novel issues → escalate to human analysis.

UC-18. Performance Analytics and Optimization
Preconditions: Analytics access granted; data collection operational.
Main Flow:

- Admin accesses performance dashboard.
- Identifies optimization opportunities.
- Platform implements changes; tracks improvement.
  Alternative Flows:
- Data quality issues → validate and correct data.
- Privacy compliance → analytics adjusted accordingly.

UC-19. Cooperative Platform Governance
Preconditions: Local restaurants invested in platform; voting mechanisms configured.
Main Flow:

- Investors participate in governance meetings.
- Platform facilitates democratic voting.
- Approved changes implemented; profit-sharing distributed.
  Alternative Flows:
- Voting deadlocks → mediation/tie-breaking.
- Stakeholder disputes → conflict resolution protocols.

UC-20. Configure System-wide Settings
Preconditions: Admin logged in.
Main Flow:

- Adjust global parameters (delivery fees, banners, notifications).
- Save changes → system applies immediately.
  Alternative Flows:
- Invalid value entered → error displayed.

UC-21. Predictive Demand Forecasting
Preconditions: Historical data available; ML algorithms trained.
Main Flow:

- System analyzes historical and external data.

- Generates staffing/inventory forecasts.
- Recommendations provided to restaurants.

Alternative Flows:

- Accuracy issues → fallback to historical averages.
- Unexpected events → adjust in real-time.

UC-22. Quality Assurance and Health Promotion

Preconditions: Health/safety policies configured; quality ratings operational.

Main Flow:

- System highlights healthier, safer restaurants.
- Customer satisfaction tracked; quality improved over time.

Alternative Flows:

- Health concerns → temporarily remove restaurants.
- Quality disputes → objective mediation provided.

UC-23. Process a Payment

Preconditions: Customer submitted payment.

Main Flow:

- System sends request to payment processor.
- Authorization received; order finalized.

Alternative Flows:

- Declined → error displayed; retry options provided.

UC-24. Send a Notification

Preconditions: Event occurred (order ready, driver arrived).

Main Flow:

- System sends notification (push, SMS, email).

Alternative Flows:

- Notification service unavailable → queue for later.

UC-25. Authenticate a User

Preconditions: User attempting login.

Main Flow:

- Verify credentials; grant access; create session.

Alternative Flows:

- Invalid credentials → error message.
- Multiple failed attempts → lock account temporarily.

UC-26. Commission Negotiation
Preconditions: Restaurant meets thresholds; admin approval required.
Main Flow:

- Restaurant requests review; system analyzes metrics.
- Tiered commission offered; restaurant accepts.
  Alternative Flows:
- Negotiation rejected → system provides criteria for future.
- Performance decline → rate adjusts back to standard.

UC-27. Local Market Adaptation Strategy
Preconditions: Expansion to new region; market research available.
Main Flow:

- Analyze local consumer behavior; adapt promotions.
- Establish local partnerships; launch region-specific features.
  Alternative Flows:
- Cultural misalignment → rapid adjustment.
- Regulatory compliance → ensure legal requirements met.

UC-28. Sustainable Growth Management
Preconditions: Rapid user growth; scalability limits identified.
Main Flow:

- Monitor growth metrics; improve infrastructure.
- Balance automation with human service quality.
  Alternative Flows:
- Growth constraints → controlled scaling.
- Market saturation → focus on service quality.

UC-29. POS System Integration
Preconditions: Restaurant uses POS; API compatible.
Main Flow:

- Configure API integration; orders flow automatically.
- Inventory/menu synchronized in real-time.
  Alternative Flows:
- POS incompatibility → universal tablet solution.

- Integration failure → fallback to manual entry.

UC-30. Manage Third-Party API Integration
Preconditions: External services used (payment, SMS).
Main Flow:

- Monitor API status; handle authentication; reconnect if failed.
  Alternative Flows:
- Persistent connection failure → log error; notify admin.

What's missing in 1a1

Project 1a1 mainly focused on the basics of a food delivery system, covering customers, staff, admin, and simple workflows such as browsing, ordering, preparing, and delivering food. However, it missed several important aspects. Key stakeholders like delivery drivers, external systems, and regulators were not considered, limiting the scope of interactions. Advanced, real-world features such as refunds, promotions, loyalty programs, scheduling, route optimization, and integrations with third-party services were also absent. In addition, the document lacked stakeholder-to-use case traceability, visual use case diagrams, and non-functional requirements such as performance, security, reliability, and accessibility. Finally, all use cases were presented as a flat list without prioritization into core, supporting, or advanced features. Overall, Project 1a1 presented only a minimal working model of a food delivery app, but not the depth, complexity, and professionalism expected in a complete system design.

Reflection on LLM Use Case Reports

While exploring the expansion of food delivery system use cases, I compared outputs from NotebookLM and Claude. NotebookLM provided a **careful, structured review** of sources, organizing use cases clearly by stakeholder role and highlighting practical gaps. Its approach emphasized **realistic, usable features** and ensured all key actors, including delivery drivers, were considered. However, the report was less engaging in style, focused mostly on basic functions, and offered fewer details on exception handling or backup flows. Additionally, NotebookLM struggled with **loading large amounts of data** and had slower response times, even for simple queries.

Claude, in contrast, approached the task **holistically**, incorporating additional stakeholders and exploring broader system interactions, including supply chain partners, IoT devices, social media, and regulatory systems. While this led to **technically advanced suggestions**, some features, such as blockchain and IoT integrations, did not clearly address the **core business or**

user needs, illustrating the importance of focusing on practical impact rather than just technological possibilities.

Overall, NotebookLM was useful for **methodically expanding existing use cases**, whereas Claude encouraged **out-of-the-box thinking** and broader system perspectives. Both tools complement human expertise, but their outputs differ in **practicality versus innovation**.