

Use Case Report — GoodFoods Reservation Agent

Goal

Long Term Goal:

Build a production-grade conversational reservation assistant that automates search, recommendations, and slot-based booking for GoodFoods. Reduce operational load, optimize table utilization, and create a scalable, data-driven booking ecosystem.

Success Criteria:

- 80% reservation completion rate
- <2% slot conflicts
- Reduce phone bookings by 60%
- 15% increase in booking conversions
- Reliable data persistence

Use Case (100 words):

GoodFoods customers want a fast, conversational way to find restaurants and reserve tables. The AI agent captures preferences such as cuisine, location, time slot, and party size, recommends the best matches, checks real-time availability, and confirms bookings. The system uses an LLM orchestrator to parse intent and call backend tools for search, availability checking, and booking. Booked slots are removed from the slot database and stored in a bookings JSON. Streamlit visualizations display analytics such as cuisine trends and ratings. This solution improves customer experience, reduces staff load, and drives better operational decisions.

Key Steps (Bot Flow):

1. User starts a query.
2. Agent collects details.
3. Agent recommends restaurants.
4. User chooses a restaurant + slot.
5. Agent checks availability.
6. Agent books the slot.
7. Agent updates database.

8. Agent shows confirmation.

Bot Features:

- Natural-language restaurant search
- Slot-based booking system
- Persistent JSON datastore
- Optional Grok LLM integration
- Analytics dashboard
- Tools: search_restaurants, get_availability, book_slot
- Difficulty:
- Green: Search, Analytics, Chat history
- Yellow: LLM orchestration
- Red: Multi-chain real-time scaling
- Integrations: Streamlit, JSON DB, Cloudflare/ngrok tunnel

Scale-up Strategy:

- Phase 1: Internal testing
- Phase 2: Rollout to all branches
- Phase 3: Add CRM/POS sync
- Phase 4: Multi-chain support

Key Challenges:

- Slot synchronization
- LLM reliability
- Real-time data consistency
- Avoiding double bookings