NomNomSafe

Sprint 1 Presentation

ASE 485 – Capstone Project

Jeff Perdue

What is NomNomSafe?

A platform for restaurants to own their allergen data

- Restaurants manage menus that carry **real safety implications**
- Allergen information is often incomplete, inconsistent, or out of date
- NomNomSafe treats menu data as **living, safety-relevant information** — not static text

Who it serves

- **Business users** (restaurant owners/staff): clear ownership of menu and allergen data
- **Customers** (future scope): reliable, structured allergen information

Sprint 1 Goals

Four features. One focus: a stable MVP.

Feature	Goal
1. Archive Non-MVP Features	Remove complexity — scope the app clearly
2. Front-End Refactor	Improve structure without changing behavior
3. Search, Filter & Sort	Make item management faster and more precise
4. Improved Business Onboarding	Make first-time setup clear and validated

Each feature was driven by actor–goal requirements documented before work began.

Feature 1: Archive Non-MVP Features

Goal: reduce complexity before adding features

What was archived

- **Admin / User Maintenance** — multi-user per business not in MVP scope
- **Menu Item Swapping** — multiple menu item management not in MVP scope
- **Add / Delete Menus** — multiple menu support stubbed, not active

Why it matters

- Active code only reflects what users can actually use
- Archived code is retained, labeled, and clearly inactive
- Makes refactoring and extension safer

Feature 2: Front-End Refactor

Goal: improve structure without changing behavior

Principles applied

- **SOLID** — Single Responsibility, Open/Closed, Dependency Inversion throughout component and utility design
- **Facade Pattern** — API layer (`api/index.js`) is a single source of truth for all HTTP calls; components depend on abstraction, not raw `fetch()`
- **Strategy Pattern** — validation rules in `formValidation.js` are pluggable; forms compose them rather than hardcode logic
- **Chain of Responsibility** — `ProtectedRoute` guards are small, single-purpose functions chained in sequence

Why it matters

- Technical debt removed *before* Sprint 2 feature work begins
- New features only need to update one file to change an endpoint or add a validation rule
- Each piece has one reason to change — easier to test, extend, and maintain

Feature 3: Search, Filter & Sort

Goal: help business users find and manage items quickly

What was delivered

- **Search** across item name, description, ingredients, and allergen names
- **Filter** by availability and by allergen (include or exclude)
- **Sort** by category, name (A–Z / Z–A), or price (low–high / high–low)
- **Category tabs** with item count badges
- Filters combine predictably; active filter badge shown when filters are on
- Sort preference persists across sessions

Feature 4: Improved Business Onboarding

Goal: make first-time setup clear, validated, and guided

Two-step flow

- **Step 1** — Business info: name, address, website
  - Optional "Find your business" search using Google Places (prefills fields)
- **Step 2** — Allergens & dietary accommodations (optional)
  - Pre-select allergens always present; mark dietary options offered

Additional improvements

- Required fields clearly indicated; validation before advancing
- **Guided tour** on first login — highlights key areas once per user
- Sign-up: password rules shown clearly, specific error messages, data preserved on failure

Sprint 1 Metrics

Sprint 1 is complete.

Metric	Result
Features completed	**4 / 4**
Actor–goal requirements met	**22 / 22**
Burndown rate	**100%**

Requirements breakdown

- R1.1–R1.5 — Archive Non-MVP (5 requirements)
- R2.1–R2.5 — Front-End Refactor (5 requirements)
- R3.1–R3.6 — Search, Filter & Sort (6 requirements)
- R4.1–R4.6 — Improved Business Onboarding (6 requirements)

MVP App State

What the app delivers today

- Business users can **sign up, log in, and onboard** their restaurant
- Menus can be **created and named**
- Menu items can be **added, edited, deleted, and duplicated**
- Items can be **tagged with allergens** from a structured list
- Items can be **searched, filtered by allergen, and sorted** across a full menu
- The app is **stable, functional, and ready for Sprint 2 feature work**

This is the business-side foundation. Everything built in Sprint 2 sits on top of it.

Sprint 2: Three Tracks

What comes next

Track	Focus	Weeks
Track A	Menu Management UX + AI-Assisted Import	10–13
Track B	Food Allergen Ontology Research (Phases 0–2)	10–14
Track C	Celebration of Student Research (April 23)	14–15

Sprint 2 runs **Weeks 10–15 (March 16 – April 26)**.

Track A: AI-Assisted Menu Import

The problem it solves

Entering a full menu manually is time-consuming.

How it works

```
Upload file / Paste URL
        ↓
  LLM parses text into structured items
        ↓
  Business user reviews: edit any field, confirm allergens
        ↓
  Save approved items to the database
```

Key design principle

AI surfaces **suggestions**. The business user **confirms**. Allergen data is never saved without explicit review — the human remains responsible.

Track B: Allergen Ontology Research

The core problem with AI and allergen safety

Current language models reason unsafely about allergens:

- "*It doesn't list peanuts, so it's probably safe.*"
- "*It's grilled, so cross-contact isn't an issue.*"
- "*No mention of nuts in this dish.*"

These inferences are not structurally justified, they treat **missing information as evidence of absence**.

What we're building

A structured **Exposure-State Framework** that requires positive evidence before declaring anything safe. Four possible states for any food item + allergen pair:

ConfirmedPresent · PotentiallyPresent · ConfirmedAbsent · Unknown

The Safety Inference Principle: a dish may be called safe only when exposure state = *ConfirmedAbsent*.

Sprint 2 Schedule & Milestones

Week	Dates	Milestone
10	Mar 16–22	Sprint 2 scope documented; LLM API access confirmed; Phase 0 data collection underway
11	Mar 23–29	Server parses uploaded files and returns structured items; Phase 0 deliverables complete
12	Mar 30–Apr 5	End-to-end file import works; URL parsing with fallback; Phase 1 ontology draft started
13	Apr 6–12	Import feature functionally complete; Phase 1 ontology fully documented
14	Apr 13–19	Poster and slides draft ready; Phase 2 ontology drafted; app demo-ready

Final Delivery: April 23

NKU Celebration of Student Research

April 23, 2026 — NKU Celebration of Student Research

Poster

The poster communicates the full arc of the project to a broad audience:

- **System overview** — what NomNomSafe is and who it serves
- **Import & parsing flow** — how AI assists menu import without replacing human review
- **Allergen safety model** — the exposure-state framework and why it matters

Sprint 1 Complete. Sprint 2 Underway.

Where we are

- Four features delivered, 22 requirements met, 100% burndown
- The MVP is stable and ready for Sprint 2 work
- Sprint 2 scope is defined, requirements are written, milestones are set

Where we're going

- AI-assisted menu import with human review
- A research-backed allergen reasoning framework
- Poster presentation at NKU's Celebration of Student Research — April 23

Questions or discussion?