

NomNom-HW 4

Team Info

- Team Name
 - *NomNom*
 - Slogan
 - Peace of mind: on every plate, every place.
 - Goal
 - Our goal is to provide a simple and user-friendly web-based application that allows restaurants to store, modify, and display accurate and up-to-date dietary information about each of their menu items.
-

Sales Pitch

- NomNom Safe helps restaurants enhance their reputation as a business that cares about their entire customer base — including those with dietary restrictions — by providing a simple and user-friendly web-based application that stores and displays accurate and up-to-date dietary information about each of their menu items. NomNom Safe allows business users to quickly and easily import, export, input, verify, and modify information about menus, disclaimers, menu items, and the restaurant itself.
-

Team Members

- Anna Dinius (Team leader)
 - Email: dinius1@mymail.nku.edu
 - Repo: <https://github.com/Anna-Dinius/ase-285-team-project>
- Jeff Perdue

- Email: perduej7@mymail.nku.edu
 - Repo: <https://github.com/jeffreyperdue/ase-285-team-project>
 - Pamela Pepper
 - Email: pepperp1@mymail.nku.edu
 - Repo: <https://github.com/echocall/ase-285-team-project>
 - Owen Newberry
 - Email: newberryo1@mymail.nku.edu
 - Repo: <https://github.com/owen-newberry/ase-285-team-project-Owen>
-

Project Info

Rules

- No Surprises
 - Divide and Conquer
 - Start early and finish early
 - Buy, don't build
 - Comment your code
 - Provide descriptions for functions
 - Provide clarification as necessary
 - CC all team members on emails to the professor regarding the project.
 - One person will take notes of what is discussed in a meeting for those who are not there.
 - The note-taker for the meeting should be available to make time to answer questions for those who could not make it.
 - Penalty in the form of low peer review points when anyone breaks the team rules.
-

Tools

Team

- GitHub
- MongoDB
- Visual Studio Code
- HTML
- CSS
- JS
- Nodejs
- React
- Notion
- Teams - Communication
- Canvas
- Google Docs

Individual

- WebStorm
 - Mongoose
-

Product Backlog

Theme: Web-based allergen-aware restaurant menu manager

Anna

- ☐ Users can sign up and log in
- ☐ Authorized users can log out
- ☐ Authorized users can edit their account information

- ☐ Admin users can authorize other users

Jeff

- ☐ Authorized users can create/edit/delete restaurant info
- ☐ Authorized users only have access to specific restaurants
- ☐ Create/edit/delete menus are tied to a restaurant
- ☐ Display menus with associated menu items

Pam

- ☐ Create/edit/delete menu items
- ☐ Display menu items on the restaurant's menu
- ☐ Import/export menu items using CSV, JSON, XML, or PDF

Owen

- ☐ Tag menu items with allergens from a preset list
- ☐ Filter menu items by allergen
- ☐ Generate allergen disclaimer based on selection or tagged items

Jeff & Pam

- ☐ Import menu/menu items using CSV, JSON, XML, or PDF
 - ☐ Export menu/menu items using CSV, JSON, XML, or PDF
-

Sprint 1 (March 24 – April 6) – Frontend Development Only

Goal: Have all major UI components built and styled (no backend yet)

Anna

- Authentication Module
 - ☐ Design and build Login page UI

- ☐ Design and build Signup page UI
- ☐ Build Logout button UI
- ☐ Design Edit Account page UI
- Admin Module
 - ☐ Design and build admin dashboard page UI

Pam

- ☐ Build Create Menu Item page (static form)
- ☐ Build Edit Menu Item page (static form)
- ☐ Build Menu Item Display page (list/table UI)

Jeff

- ☐ Build Restaurant Creation page UI
- ☐ Build Restaurant Edit page UI
- ☐ Build Menu Creation/Edit pages UI

Owen

- ☐ Build Tagging UI (dropdown or checkboxes)
- ☐ Build Filter UI (search/filter panel)
- ☐ Build Disclaimer UI (text area or selection modal)

Sprint 2 (April 7 – April 20) – Backend Development & Integration

Goal: Connect frontend to backend with MongoDB, implement business logic

Pam

- ☐ Implement Menu Item class (Mongoose schema)
- ☐ Add CRUD functionality to forms/pages

- ☐ Connect search bar to backend query
- ☐ Test and validate menu item routes

Anna

- Authentication Module
 - ☐ Implement user registration/login/logout logic
 - ☐ Add session/token management
 - ☐ Restrict access to pages based on auth state
- Admin Module
 - ☐ Add user role authorization logic

Jeff

- ☐ Implement Restaurant and Menu schemas
- ☐ Create backend routes for Restaurant and Menu CRUD
- ☐ Connect frontend forms to backend logic

Owen

- ☐ Implement tagging system for menu items
- ☐ Enable filtering logic based on allergens
- ☐ Build backend logic to generate disclaimers

Jeff & Pam

- ☐ Create CSV/JSON import/export utilities
- ☐ Connect import/export buttons to actual logic

Testing

- ☐ Write tests for core flows (menu item CRUD, auth, allergens)
 - ☐ Perform integration testing (UI + DB)
-

Requirements

Anna

- Authentication Module
 - As a new user, I should be able to create an account with my business email so I can create a profile for my restaurant(s).
 - As a returning user, I should be able to log in with my business email and password so I can access my restaurant's page.
 - As an authorized user, I should be able to view the restaurants I have access to so I can make changes to them.
 - As an authorized user, I should be able to delete my own account.
- Admin Module
 - As an administrator, I should be able to control which users have access to view and modify the information for my restaurants so I can share responsibilities.
 - As an administrator, I should be able to promote certain users to administrator status so I can share responsibilities.

Owen

- Tagging System
 - As an authorized user, I should be able to tag menu items as having certain allergens from a pre-created list of allergens so I don't have to manually type them in.
- Disclaimer Generation
 - As an authorized user, I should be able to choose a default disclaimer for my restaurant about the allergens present in my kitchen so that I am legally covered, my customers know what may be present, and I don't have to write one myself.
 - As an authorized user, I should be able to add a custom disclaimer to my restaurant about the allergens present in my kitchen so that I am legally

covered and my customers know what may be present.

- Filtering
 - As an authorized user, I should be able to filter the menu items based on allergens so I can quickly find and modify them.

Pam

- Menu Item CRUD
 - As an authorized user, I should be able to create new menu items for my restaurants so I can keep the menu accurate and up-to-date.
 - As an authorized user, I should be able to edit existing menu items for my restaurants so I can keep the menu accurate and up-to-date.
 - As an authorized user, I should be able to delete existing menu items for my restaurants so I can keep the menu accurate and up-to-date.
- Menu Item Search
 - As an authorized user, I should be able to search for a specific menu item so I can find it quickly if I need to alter it.

Jeff

- Menu CRUD
 - As an authorized user, I should be able to create menus associated with the restaurant(s) I have access to so I can keep the information accurate and up-to-date.
 - As an authorized user, I should be able to edit menus associated with the restaurant(s) I have access to so I can keep the information accurate and up-to-date.
 - As an authorized user, I should be able to delete menus associated with the restaurant(s) I have access to so I can keep the information accurate and up-to-date.
- Restaurant CRUD

- As an authorized user, I should be able to add delete information about the restaurant I have access to alter so I can keep the information up to date.
- As an authorized user, I should be able to edit information about the restaurant I have access to alter so I can keep the information up to date.
- As an authorized user, I should be able to delete information about the restaurant I have access to alter so I can keep the information up to date.

Jeff & Pam

- Import/Export Utilities
 - As an authorized user, I should be able to import menu information from a file so I can upload my old excel sheets into the new system.
 - As an authorized user, I should be able to export menu information to a file so I can have a copy of all my restaurant's menu item data.
-

Tests Plan

- Unit tests: each person is responsible for the unit tests for their modules.
 - Integration tests: team members work together on integration tests for the modules being integrated.
 - Regression tests: will be created and run by the team members who created the modules being tested.
-

Meeting Schedule

- 9:00 AM - 10:00 AM every Friday
 - 8:30 PM - 9:30 PM every Sunday
-

Milestones:

- Complete UI/Front-End (1st Sprint)
- Back-End Completed/Fully integrated with Front-End (2nd Sprint)
- All tests passed, bugs fixed, ready for final submission (3rd Sprint)

Deadlines

- Each team member's Sprint contribution is due Friday before end of Sprint.
 - Sprint 1: April 4
 - Sprint 2: April 18
 - Final Submission: 3 days before Canvas due date
-

Team Member Schedules

Anna

<u>Day</u>	<u>Time</u>	<u>Commitment</u>
M	10:00 am - 1:00 pm	Work (no commute necessary)
	2:00 - 3:15 pm	ASE 285
T	10:50 am - 12:05 pm	HNR 311
	1:40 - 2:55 pm	CSC 364
	5:30 - 6:30 pm	BJJ
W	10:00 am - 1:00 pm	Work (no commute necessary)
	2:00 - 3:15 pm	ASE 285
R	10:50 am - 12:05 pm	HNR 311
	1:40 - 2:55 pm	CSC 364
	6:00 - 10:00 pm	Newman
F	1:00 - 5:00 pm	Work (not including commute)

Sa	10:30 - 11:30 am	BJJ
Su	8:30 - 9:30 am	Mass

Jeff

<u>Day</u>	<u>Time</u>	<u>Notes</u>
M	8:30 am - 11 am	
	2 pm - 3:15 pm	
Tu	9 am - 12 pm	Flexible
W	8:30 am - 11 am	
	2 pm - 3:15 pm	
	4 pm - 10:30 pm	Work (not counting commute)
Th	3:30 pm - 10:30 pm	Work (not counting commute)
F	8:30 am - 11 am	Work (not counting commute)
	3:30 pm - 10:30 pm	
Sa		Work (not counting commute)
Su	9:30 am - 5:00 pm	Work (not counting commute)

9:30 am - 5:00
pm

Owen

<u>Day</u>	<u>Time</u>	<u>Commitment</u>
M	12:00 - 12:50 pm	Class
	2:00 - 3:15 pm	ASE 285
T	9:25 - 10:40 am	Class
	12:15 - 1:30 pm	Class
	4:00 - 8:15 pm	Work
W	2:00 - 3:15 pm	ASE 285
R	9:25 - 10:40 am	Class
	12:15 - 1:30 pm	Class
	4:00 - 8:15 pm	Work
F		
Sa		
Su	6:00 - 8:00 pm	Busy

Pam

Day	Time	Commitment
-----	------	------------

M	10:00 - 10:50 am	ANT 309
	2:00 - 3:15pm	ASE 285
T	1:40 - 2:55pm	ASE 485: Every two weeks
	Online Study	CSC 364
W	10:00 - 10:50 am	Ant 309
	2:00 - 3:15pm	ASE 285
Th	Online Study	CSC 364
F	10:00 - 10:50 am	ANT 309
	7:00 - 11:00pm	PF2E
Sat	Online Study	CSC 364
Sun	2:00 - 6:00 pm	PF2E

Significant Dates:

- March 24 ASE 485 Capstone 2nd Prototype
- March 24 ASE 285 Homework Assignment 4
- March 30 CSC 364 Assignment 5
- March 30 CSC 364 Quiz 5
- March 31 ANT 309 - Culture Paper - Sociopolitical Organization
- April 7 ASE 485 Capstone Analysis and results
- April 14 ANT 309 - Culture Paper - Religion and Expressive Culture
- April 21 ASE 485 Capstone Revised Draft/Prototype
- May 5 ASE 485 Capstone Final Deliverables
- May 3 ASE 285 Homework Assignment 8
- May 9 ANT 309 - Final Culture Paper

