

Capstone Project

Week 6 Progress Report

2/16/2026 - 2/22/2026

Commit range:

- c792cc59449e4ad022cfbca7fb9f395dcdaac6a2 →
1247fc9240b6d116432320f682fe8bcef9a5cd6f

See [project-statistics-detailed.md](#) for LoC and other project statistics.

Summary

This week focused on:

- Large client-side UI refactor and component consolidation
- Adding several reusable UI primitives and app-level context
- Hook-based refactors for auth and account flows

Key Outcomes

- Large client-side refactor of authentication & account flows (renames and component consolidation): `SignUp` → `AuthFormSwitcher` , `Password` → `PasswordFormField` , extracted `FormField` , consolidated `GetAuthForm` and sign-in/up forms
- New reusable UI primitives: `LoadingSpinner` , `Modal` , `Toast` + `ToastContext` , `SaveButton` , `PageLayout` , `Nav` and breadcrumb support
- Improved allergen flows: moved `AllergenList` → `GenerateAllergenList` , added `AllergenPicker` and `AllergenCheckbox` , introduced `useAllergens` hook
- Numerous hooks & utilities added for consistent form and auth handling (`useAuthActions` , `useAccountDetails` , `useBreadcrumbs` , `parseFullName` , `splitName`)

New Files (Server)

- server/src/controllers/categoryController.js
- server/src/routes/categoryRoutes.js
- server/src/services/categoryService.js

New Files (Client)

- client/src/components/common>LoadingSpinner/LoadingSpinner.jsx
- client/src/components/common/Modal/Modal.jsx
- client/src/components/common/Toast/Toast.jsx and ToastContainer.jsx
- client/src/components/common/FormField/FormField.jsx
- client/src/components/common/AllergenPicker/AllergenPicker.jsx
- client/src/components/common/ManageCategoriesModal/ManageCategoriesModal.jsx
- client/src/context/ToastContext.jsx
- Auth form reorganized into AuthFormSwitcher and GetAuthForm subcomponents
- Hooks: useAllergens , useAuthActions , useAccountDetails , useBreadcrumbs , useEditName

Major File Modifications (Server)

- Small fixes in `businessController` helpers

Major File Modifications (Client)

- Large refactor of auth/account area: replaced `EditLoginInfo` with hook-based flows, reorganized auth components into `AuthFormSwitcher` and `GetAuthForm` subcomponents
- Added global UI primitives (Modal, Toast) and a `ToastContext` for app-wide notifications
- Updated `App.jsx` and routing to integrate the new layout, breadcrumbs, and nav components

Tests

Needed Next

- Unit tests for `categoryService` and category routes
- Client tests for new components: `FormField`, `AllergenPicker`, and `LoadingSpinner` flows

Rationale

Why these changes?

- **Client consolidation & reusability**
 - Centralize common UI primitives (`Modal` , `Toast` , `FormField` , `LoadingSpinner`) to reduce duplication and simplify future screens
- **Move to hook-based auth flows**
 - Hooks provide clearer state flows and easier reuse across sign-in, sign-up, and account edit screens

Files Touched (Concise)

Summary

- 106 files changed, ~6007 insertions(+), ~1728 deletions(-) across client and server

Notable Added (client/server)

- Client: `LoadingSpinner`, `Modal`, `Toast`, `FormField`, `AllergenPicker`,
`ManageCategoriesModal`, `AuthFormSwitcher` components; `ToastContext`; multiple
hooks and utils
- Server: `categoryController`, `categoryService`, `categoryRoutes`

Removed / Archived

- Deprecated auth components and styles (moved/rename into `AuthFormSwitcher`

/ `GetAuthForm`)

How to Run / Verify Locally

1. Install dependencies

```
npm install
```

2. Run

```
npm run dev
```

Manual Verification Checklist

- Auth flows show updated sign-in / sign-up forms via `AuthFormSwitcher`
- Manage categories modal opens and persists category changes
- Toast notifications display during save/error operations
- LoadingSpinner appears during async operations

Notes & Next Steps

- Update client & server tests to reflect renamed components and schema updates
- Consider memoizing allergen data for performance in large menus

Completion Summary

This week delivered a major client-side UI consolidation and improved server/client alignment. The repo contains broad refactors across auth and account flows that improve reusability and UX while laying groundwork for robust testing.

Metrics

🔥 Week 6 Burndown Rates

- 16/16 Week 6 requirements completed
 - 100% total
 - ~14% per day

Metrics (continued)

🔥 Sprint 1 Burndown Rates

- 3/4 Sprint 1 features completed
 - 75% total
 - 25% per week
 - ~3.6% per day
- 30/41 Sprint 1 requirements completed
 - ~73% total
 - ~24% per week
 - ~3.5% per day