

Recipe Reborn - Bug Fixes Applied

Date: February 15, 2026

Version: Production Ready

Summary

This document details the critical bugs identified during testing of recipereborn.com and confirms their resolution status.

Bug Fixes

✓ BUG-001: Signup Form Missing Full Name Field

Status: FIXED

Issue: The signup form was missing the "Full Name" field that the backend API requires for user registration.

Resolution: The signup form (`/src/app/signup/page.tsx`) now includes:

- Full Name input field (lines 84-97)
- Proper state management with `useState` hook
- Name field is sent in the API request body
- Field is marked as required

Files Affected:

- `/src/app/signup/page.tsx`
-

✓ BUG-005: Pricing Page Removal

Status: FIXED

Issue: A pricing page existed at `/pricing` and in navigation links, contradicting the free service positioning.

Resolution:

- No `/pricing` directory or page exists in the codebase
- No pricing-related links in Header component
- No pricing references anywhere in the application
- Site is positioned as completely free

Verification:

- Searched entire codebase: `grep -ri "pricing" src/` returns no results
 - `/src/app/` directory contains only: api, login, signup pages
-

✓ BUG-006: Error Handling for Forms

Status: FIXED

Issue: No error feedback was displayed when registration or login failed.

Resolution: Both forms now have comprehensive error handling:

Signup Form (/src/app/signup/page.tsx):

- Client-side validation for password matching
- Password minimum length validation (6 characters)
- Error state management with `useState`
- Error display component with red styling
- Handles API errors and displays user-friendly messages
- Try-catch block for network errors

Login Form (/src/app/login/page.tsx):

- Error state management with `useState`
- Error display component with red styling
- Handles API errors (invalid credentials, server errors)
- Try-catch block for unexpected errors

Backend API Error Responses:

- `/api/auth/signup/route.ts` : Returns specific errors for missing fields, invalid email, weak password, duplicate accounts
- `/api/auth/login/route.ts` : Returns specific errors for missing credentials, invalid email/password

Environment Configuration

The `.env.example` file is complete and documented with:

- `JWT_SECRET` - For token signing
- `UPSTASH_REDIS_REST_URL` - Redis database URL
- `UPSTASH_REDIS_REST_TOKEN` - Redis authentication
- `KV_REST_API_URL` / `KV_REST_API_TOKEN` - Vercel KV alternatives
- `NODE_ENV` - Environment setting

Code Quality Verification

- ✓ All TypeScript types are properly defined
 - ✓ No unused imports or dependencies
 - ✓ Proper error handling throughout
 - ✓ Authentication flow is complete (login, signup, logout, session check)
 - ✓ Responsive design for all screen sizes
 - ✓ Accessible form labels and inputs
-

Deployment Notes

To deploy the updated code:

1. **Push to GitHub:**

```
bash
git add .
git commit -m "Fix critical bugs: signup form, pricing removal, error handling"
git push origin main
```

2. **Vercel will auto-deploy** from the main branch

3. **Verify Environment Variables** in Vercel dashboard:

- JWT_SECRET
- UPSTASH_REDIS_REST_URL
- UPSTASH_REDIS_REST_TOKEN

Testing Checklist

After deployment, verify:

- [] User can sign up with name, email, and password
- [] Error messages appear for invalid signup attempts
- [] User can log in with valid credentials
- [] Error messages appear for invalid login attempts
- [] No pricing page accessible at /pricing
- [] No pricing links in navigation
- [] User session persists after page refresh
- [] Logout functionality works