

Chef Doggo - Phase 2 Handoff Document

Date: December 27, 2024

Status: Phase 1 Complete ✓ | Phase 2 Database Schema Complete ✓ | Ready to Build Phase 2 Features

Latest Commit: Phase 2 database schema update

Total Commits: 17



Quick Start for New Conversation

Copy and paste this to start Phase 2:

Continue building Chef Doggo Phase 2 from checkpoint "Phase 1 Complete".

Project path: /home/ubuntu/chef_doggo

GitHub: <https://github.com/Bharv1122/chef-doggo>

Phase 2 Prompt: /home/ubuntu/Uploads/phase-2-enhancement-prompt.md

Database schema has been updated with all Phase 2 tables (meal_plans, nutrition_logs, health_observations, progress_photos, milestones, ingredients_safety, supplement_interactions) **and** DogProfile model enhanced with holistic fields (tcvmConstitution, tcvmThermalNature, ayurvedicDosha, conditionDiet).

Prisma client generated **and** database pushed successfully.

Now implement Phase 2 features starting with:

1. Holistic dog profile form enhancement
2. Enhanced recipe generation with holistic analysis
3. Weekly meal planner
4. Nutrition tracker
5. Progress photos
6. Remaining Phase 2 features

Follow the detailed specifications **in** phase-2-enhancement-prompt.md.



Project Overview

What is Chef Doggo?

Chef Doggo transforms commercial dog kibble into personalized homemade recipes, integrating:

- **Veterinary Nutrition Science** (AAFCO standards)
- **Holistic Approaches** (TCVM and Ayurveda)
- **Safety Features** (toxic ingredient validation, medication checks)
- **5-Tier Disclaimer System** with database tracking

Tech Stack

- **Framework:** Next.js 14 (App Router)
- **Database:** PostgreSQL (Hosted by Abacus AI)

- **ORM:** Prisma 6.7.0
- **Auth:** NextAuth 4.24.11
- **Styling:** Tailwind CSS + shadcn/ui
- **AI:** OpenAI GPT-4 Vision (via Abacus AI API)
- **Package Manager:** yarn

📁 Important Paths

```

/home/ubuntu/chef_doggo/
├── nextjs_space/
│   ├── app/
│   │   ├── api/
│   │   │   ├── my-dogs/
│   │   │   ├── generate/
│   │   │   ├── recipes/
│   │   │   ├── auth/
│   │   └── components/
│   │       └── forms/dog-profile-form.tsx
│   DATES)
│   ├── lib/
│   │   ├── tcvm.ts
│   │   ├── ayurveda.ts
│   │   ├── safety-validation.ts
│   │   ├── recipe-generator.ts
│   │   ├── nutrition-utils.ts
│   │   ├── medication-interactions.ts
│   │   ├── recipe-utils.ts
│   │   ├── cost-estimation.ts
│   │   └── prisma/
│   │       └── schema.prisma
│   └── public/
│       └── chef-doggo-logo.webp
└── PHASE_2_HANDOFF.md
/home/ubuntu/Uploads/
    ├── phase-2-enhancement-prompt.md
    ├── chef-doggo-project-bible.md
    └── original-design-reference.md

```

Project root
Next.js application
App Router pages
API routes
Dog profile management
Recipe generation
Saved recipes
Login/signup pages
React components
Dog profile form (NEEDS PHASE 2 UP-
Recipe display
shadcn components
Utility functions
TCVM integration
Ayurveda integration
P0 safety validation
Recipe generation
Nutrition calculations
Medication checker (25+ meds)
Batch scaling, shopping list
Cost calculator
✓ UPDATED WITH PHASE 2 TABLES
Logo
This file
Full Phase 2 specifications
Original project vision
Design guidelines

✓ Phase 1 Complete - What's Already Built

Core Features

- ✓ Landing page with Chef Doggo branding
- ✓ User authentication (email/password)
- ✓ Dog profile system (basic info, health, dietary needs, medications)
- ✓ Recipe generation with AAFCO compliance
- ✓ AI-generated recipe photos (GPT-4 Vision)
- ✓ Saved recipes functionality
- ✓ Affiliate links (Chewy integration)

- 5-tier disclaimer system with database tracking
- Mobile responsive design
- Cost estimation for recipes

Advanced Features (Phase 1B/1C)

- Kibble label scanning (AI Vision)
- TCVM integration (50+ ingredients, constitution determination)
- Ayurveda integration (dosha system, conflict detection)
- P0 Safety Validation (40+ toxic ingredients, dangerous combinations)
- Enhanced medication interactions (25+ medications, 10 categories)
- Recipe utilities (batch scaling, shopping lists, export)

Database Tables (Phase 1)

- `users` - User accounts
- `accounts` - OAuth accounts
- `sessions` - Active sessions
- `verification_tokens` - Email verification
- `dog_profiles` - Dog information
- `recipes` - Generated recipes
- `disclaimer_acknowledgments` - Disclaimer tracking



Phase 2 Database Schema - COMPLETED

New Tables Added

Meal Planning

```

model MealPlan {}  

  id          String  @id @default(cuid())  

  userId      String  

  dogProfileId String  

  name        String  

  startDate   DateTime  

  endDate     DateTime?  

  entries     MealPlanEntry[]  

}  
  

model MealPlanEntry {}  

  id          String  @id @default(cuid())  

  mealPlanId  String  

  date        DateTime  

  mealType    String  // breakfast/lunch/dinner/snack  

  recipeId    String?  

  notes       String?  

}

```

Nutrition Tracking

```

model NutritionLog {
    id          String  @id @default(cuid())
    userId      String
    dogProfileId String
    date        DateTime
    recipeId    String?
    customEntry String?
    portionSize Float
    treats       String?
    waterIntake Float?
    notes        String?
}

model HealthObservation {
    id          String  @id @default(cuid())
    userId      String
    dogProfileId String
    date        DateTime
    energyLevel Int?    // 1-5 scale
    stoolQuality Int?    // 1-5 scale
    coatCondition String? // excellent/good/fair/poor
    symptoms    Json    @default("[]")
    weight      Float?
    notes        String?
}

```

Progress Tracking

```

model ProgressPhoto [] {
    id          String  @id @default(cuid())
    userId      String
    dogProfileId String
    photoUrl    String
    category    String // full-body/coat-closeup/face
    notes       String?
    takenAt     DateTime
}

model Milestone [] {
    id          String  @id @default(cuid())
    userId      String
    dogProfileId String
    title       String
    description String?
    date        DateTime
}

```

Reference Data

```

model IngredientSafety {
  id          String    @id @default(cuid())
  name        String    @unique
  safetyRating String   // safe/caution/toxic
  explanation String   @db.Text
  servingSize String?
  preparation String?
  thermalNature String? // cooling/neutral/warming/hot (TCVM)
  doshaEffect String? // which dosha it balances (Ayurveda)
  elementSupport String? // which element it supports (Five Elements)
}

model SupplementInteraction {
  id          String    @id @default(cuid())
  supplement1 String
  supplement2 String
  interactionType String // conflict/enhances/caution
  description String   @db.Text
  severity     String   // low/moderate/high
}

```

Enhanced DogProfile Model

```

// NEW FIELDS ADDED:
nutritionPhilosophy String? // Balanced/Raw-BARF/Ancestral/Whole-Food/Rotational/
Functional
tcvmConstitution String? // wood/fire/earth/metal/water
tcvmThermalNature String? // yang-hot/yin-cold/balanced
ayurvedicDosha String? // vata/pitta/kapha
conditionDiet String? // anti-inflammatory/ketogenic/renal/cardiac/diabetic/
elimination

```

Relations Added

```

// User model now has:
mealPlans      MealPlan[]
nutritionLogs NutritionLog[]
healthObservations HealthObservation[]
progressPhotos ProgressPhoto[]
milestones     Milestone[]

// Recipe model now has:
mealPlanEntries MealPlanEntry[]
nutritionLogs NutritionLog[]

```

Status: Schema updated, Prisma client generated, database pushed successfully

Phase 2 Requirements Summary

12 Major Features to Build

1. Holistic Nutrition Options ★ HIGH PRIORITY

Add to dog profile form:

- Nutrition Philosophy Selection (6 options: Balanced/Raw-BARF/Ancestral/Whole-Food/Rotational/Functional)
- TCVM Five Element Constitution (Wood/Fire/Earth/Metal/Water) with organ meat recommendations
- TCVM Food Energetics (Thermal Nature: Hot/Warming/Neutral/Cooling)
- TCVM Constitutional Types (Yang-Hot/Yin-Cold/Balanced)
- Ayurvedic Dosha (Vata/Pitta/Kapha) with breed examples and dietary guidelines
- Condition-Specific Diets (6 options with detailed modifications)

Recipe Generation Updates:

- Include holistic analysis in recipes
- Show thermal nature badges on ingredients
- Add "Holistic Notes" section

File to Update: components/forms/dog-profile-form.tsx

New Component: Create components/forms/holistic-options-tab.tsx

API Update: app/api/dogs/route.ts and app/api/dogs/[id]/route.ts

2. Weekly Meal Planner ★ HIGH PRIORITY

- Calendar view (7 days)
- Drag-and-drop recipe assignment
- Breakfast/dinner slots
- Auto-suggest protein variety
- Generate shopping list from meal plan
- Save meal plan templates
- Batch cooking suggestions

New Pages:

- app/meal-planner/page.tsx - Main planner view
- app/meal-planner/[id]/page.tsx - Individual plan editor

New API Routes:

- app/api/meal-plans/route.ts - Create/list meal plans
- app/api/meal-plans/[id]/route.ts - Get/update/delete plan
- app/api/meal-plans/[id]/shopping-list/route.ts - Generate list

Components:

- components/meal-planner/calendar-view.tsx
- components/meal-planner/recipe-selector.tsx

3. Nutrition Tracker ★ HIGH PRIORITY

Daily Feeding Log:

- Log meals (recipe or custom entry)
- Track portions, treats, water intake
- Health observations (energy, stool, coat, symptoms)
- Weight tracking with graph

Insights Dashboard:

- Calorie intake vs. target
- Protein/fat/carb breakdown over time
- Correlation alerts (e.g., "Itching after chicken")

New Pages:

- `app/nutrition-tracker/page.tsx` - Dashboard
- `app/nutrition-tracker/log/page.tsx` - Daily log entry

New API Routes:

- `app/api/nutrition-logs/route.ts`
- `app/api/health-observations/route.ts`
- `app/api/nutrition-logs/insights/route.ts`

4. Progress Photos & Timeline

- Photo upload (before/after pairs)
- Categories: full-body/coat-closeup/face
- Chronological timeline
- Weight overlay
- Milestone markers
- Transformation stories (optional public sharing)

New Pages:

- `app/progress/page.tsx` - Timeline view
- `app/progress/upload/page.tsx` - Photo upload

New API Routes:

- `app/api/progress-photos/route.ts`
- `app/api/milestones/route.ts`

5. Transition Guide Generator

- Personalized 7/10/14-day plans
- Based on current diet and sensitivity
- Day-by-day mixing ratios
- What to watch for
- Printable PDF

New Component: `components/transition-guide/generator.tsx`

New API Route: `app/api/transition-guide/generate/route.ts`

6. Ingredient Safety Database

- Searchable database of all common human foods
- Safety rating: Safe/Caution/Toxic
- Detailed explanations
- Safe serving sizes by weight
- Preparation requirements
- Thermal nature (TCVM)
- Dosha effect (Ayurveda)
- Element support (Five Elements)
- "Can my dog eat [X]?" quick check

- Voice search option
- Emergency info (symptoms, vet contacts)

New Pages:

- `app/ingredients/page.tsx` - Search interface
- `app/ingredients/[name]/page.tsx` - Detail page
- `app/emergency/page.tsx` - Emergency guide

New API Routes:

- `app/api/ingredients-safety/route.ts`
- `app/api/ingredients-safety/search/route.ts`

Seed Data: Create `scripts/seed-ingredients.ts` with comprehensive database

7. Printable Recipe Cards

- Beautiful PDF generation
- Kitchen-friendly format
- AI-generated photo included
- Ingredient checklist
- QR code linking to app

Package: Install `jspdf` or use `@react-pdf/renderer`

New API Route: `app/api/recipes/[id]/pdf/route.ts`

8. Supplement Interaction Checker

- Check supplement conflicts
- Check supplement + medication conflicts
- Dosage warnings by weight
- “Ask your vet” flags

Seed Data: Create `scripts/seed-supplements.ts`

Integration: Add to recipe generation flow

9. Recipe Variations

Quick modification buttons:

- “Make it cooling” - swap to cooling proteins/veggies
- “Make it warming” - swap to warming ingredients
- “Add joint support” - add bone broth, glucosamine foods
- “Make it low-fat” - reduce fat content
- “Boost protein” - increase protein ratio

New API Route: `app/api/recipes/[id]/variations/route.ts`

New Component: `components/recipes/variation-buttons.tsx`

10. Seasonal Recipe Suggestions

- Auto-detect user location/season
- Summer: Cooling recipes (duck, fish, cucumber)
- Winter: Warming recipes (chicken, lamb, sweet potato)
- Spring: Liver-supporting foods (Wood element)
- Fall: Lung-supporting foods (Metal element)
- Allergy season: Anti-inflammatory focus

New API Route: app/api/recipes/seasonal/route.ts

11. Breed Health Predispositions

Auto-suggest based on breed:

- Golden Retrievers → anti-inflammatory, antioxidant focus
- German Shepherds → joint support, weight management
- Bulldogs → cooling proteins, omega-3s
- Dachshunds → weight management, anti-inflammatory
- Labs → portion control, Kapha-balancing

New Utility: lib/breed-predispositions.ts

Integration: Add to dog profile form and recipe generation

12. Emergency Info Section

- Step-by-step emergency guide
- ASPCA Poison Control: (888) 426-4435
- Pet Poison Helpline: (855) 764-7661
- Nearest emergency vet locator
- Symptoms to watch for by toxin type
- When to induce vomiting vs. when NOT to

New Page: app/emergency/page.tsx



Commands You'll Need

Development

```
# Start dev server
cd /home/ubuntu/chef_doggo/nextjs_space && yarn dev

# Install new packages
cd /home/ubuntu/chef_doggo/nextjs_space && yarn add [package-name]

# Update Prisma after schema changes
cd /home/ubuntu/chef_doggo/nextjs_space && yarn prisma generate
cd /home/ubuntu/chef_doggo/nextjs_space && yarn prisma db push
```

Git Commands

```
cd /home/ubuntu/chef_doggo
git add -A
git commit -m "Your commit message"
git push origin master
```

Testing & Deployment

```
# Test the app (when ready)
test_nextjs_project with project_path=/home/ubuntu/chef_doggo

# Create checkpoint (when features complete)
build_and_save_nextjs_project_checkpoint with project_path=/home/ubuntu/chef_doggo
```

GitHub Information

Repository: <https://github.com/Bharv1122/chef-doggo>

Username: Bharv1122

Default Branch: master

Personal Access Token: [Available in conversation history - not shown for security]

Current Stats

- **Total Files:** 197+
- **Lines of Code:** ~8,000
- **Database Tables:** 11 (4 from Phase 1, 7 new in Phase 2)
- **API Endpoints:** 11+ (will grow significantly in Phase 2)
- **Safety Check Layers:** 4
- **Medications Tracked:** 25+
- **Ingredients in Databases:** 100+ (TCVM + Ayurveda + Safety)
- **Git Commits:** 17

Recommended Implementation Order

Session 1: Core Holistic Features (HIGH VALUE)

1. **Holistic dog profile form** (Foundation for everything)
 - Create new tab component
 - Update API routes
 - Test with existing dogs
2. **Enhanced recipe generation** (Uses holistic data)
 - Add holistic analysis to recipe output
 - Show thermal nature badges
 - Add “Holistic Notes” section
3. **Recipe variations** (Quick win, high impact)
 - “Make it cooling/warming” buttons
 - Uses existing TCVM/Ayurveda logic
4. **Checkpoint:** “Phase 2A: Holistic Enhancements Complete”

Session 2: Meal Planning (CORE FEATURE)

1. **Meal planner UI**
 - Calendar view
 - Recipe assignment
2. **Shopping list generator**
 - Uses existing recipe-utils.ts
3. **Checkpoint:** "Phase 2B: Meal Planning Complete"

Session 3: Tracking & Progress (USER ENGAGEMENT)

1. **Nutrition tracker**
 - Daily log entry
 - Insights dashboard
2. **Progress photos**
 - Upload interface
 - Timeline view
3. **Checkpoint:** "Phase 2C: Tracking Complete"

Session 4: Reference Data & Polish

1. **Ingredient safety database**
 - Seed database
 - Search interface
2. **Emergency info section**
3. **Printable recipe cards**
4. **Supplement checker**
5. **Transition guide generator**
6. **Seasonal suggestions**
7. **Breed predispositions**
8. **Final Checkpoint:** "Phase 2 Complete"

Important Notes

Design Consistency

- Use existing color palette from `globals.css`
- Match design language from Phase 1
- Mobile-first responsive design
- Use shadcn/ui components where possible

Data Management

- Production database is live - be careful with migrations
- Always test schema changes locally first

- Use `yarn prisma db push` (not migrate) for now

API Standards

- All routes require authentication check
- Return consistent error formats
- Use proper HTTP status codes
- Handle edge cases (no dogs, no recipes, etc.)

Testing

- Test each feature before moving to next
 - Use existing test user: `test@chefdoggo.com`
 - Run full test suite before checkpoint
-

Success Criteria for Phase 2

Phase 2 is complete when:

- Holistic options fully integrated into dog profiles and recipes
 - Meal planner with weekly calendar view works
 - Nutrition tracker logs and displays data
 - Progress photos can be uploaded and viewed in timeline
 - Transition guide generates personalized plans
 - Ingredient safety database is searchable
 - Printable recipe cards generate as PDF
 - Recipe variation buttons modify recipes correctly
 - Seasonal suggestions appear at appropriate times
 - Breed predispositions auto-populate
 - Emergency info section is accessible
 - Supplement checker validates interactions
 - All tests pass
 - Code pushed to GitHub
 - Checkpoint created
-



Next Steps

1. Start new DeepAgent conversation
 2. Copy the “Quick Start” prompt from top of this document
 3. DeepAgent will have full context to continue
 4. Work through features in recommended order
 5. Create checkpoints after each major section
-



Chef Doggo Vision

“Transform every dog’s kibble into a personalized, holistic, AAFCO-compliant homemade meal that respects both modern veterinary science and ancient healing traditions.”

Phase 1: ✓ Foundation & Core Features

Phase 2: ↲ Enhanced Holistic Platform

Phase 3: ⏳ Community & Social Features (Future)

Phase 4: 🕒 Advanced Analytics & AI (Future)

Good luck building Phase 2! 🚀🌟