Phase 1: Clean Foundation - Changes Summary

Overview

Phase 1 involved restructuring the Pet Jet website from a single Next.js application into a Turborepo monorepo. This provides a clean foundation for future growth with Strapi CMS, GoHighLevel integration, and potential admin portal.

Major Changes

1. Monorepo Structure

Before:

```
pet_jet_website/
└── nextjs_space/    # Single Next.js app
```

After:

```
pet_jet_website/
├── apps/
│   └── web/          # Next.js app (moved from nextjs_space)
├── packages/
│   ├── types/        # Shared TypeScript types
│   └── api-client/   # API utilities
└── turbo.json        # Turborepo config
```

2. Database Removal

Removed:
- Prisma ORM
- PostgreSQL database
- All database schemas and migrations
- `@prisma/client` dependency

Reason: Switching to GoHighLevel for all form submissions and lead management.

3. Form API Routes Updated

Changes:
- Removed all Prisma database operations
- Added comprehensive validation using shared utilities
- Prepared structure for GHL API integration
- Added clear TODO comments for Phase 2

Updated Routes:
- `/api/partnerships` - Partnership applications
- `/api/contact` - Contact form submissions

- `/api/product-evaluation` - Product evaluation requests
- `/api/joey-booking` - Joey Villani booking requests

4. Shared Packages Created

@pet-jet/types

Purpose: Centralized TypeScript type definitions

Contents:
- `PartnershipFormData`
- `ContactFormData`
- `ProductEvaluationFormData`
- `JoeyBookingFormData`
- `GHLContact`, `GHLOpportunity` (for Phase 2)
- `GHLAPIResponse`

@pet-jet/api-client

Purpose: Shared API utilities and clients

Contents:
- `GHLClient` class (placeholder for Phase 2)
- Validation utilities (validateEmail, validatePhone, etc.)
- Error handling utilities
- `ValidationException` class

5. Component Restructuring

Before:

```
components/
├── navigation.tsx
├── footer.tsx
├── feature-card.tsx
└── ui/
```

After:

```
components/
├── ui/              # Generic reusable components
├── layout/          # Navigation, Footer
│   ├── navigation.tsx
│   └── footer.tsx
├── feature-card.tsx # Business components
└── ...
```

Reason: Preparing for potential extraction to shared UI package in the future.

6. TypeScript Configuration

Added path mappings:

```
{
  "paths": {
    "@/*": ["./*"],
    "@pet-jet/types": ["../../packages/types/src"],
    "@pet-jet/api-client": ["../../packages/api-client/src"]
  }
}
```

7. Turborepo Configuration

Created `turbo.json`:
- Defined build, dev, lint, and clean tasks
- Set up caching strategy
- Configured task dependencies

Breaking Changes

For Developers

1. **Import paths changed for layout components:**
   ```typescript
   // Before
   import Navigation from "@/components/navigation";

// After
import Navigation from "@/components/layout/navigation";
```

1. **No database access in API routes:**
   - Forms no longer save to database
   - Phase 2 will implement GHL integration

2. **New workspace structure:**
   - Must run `yarn install` from project root
   - Can still run dev server from `/apps/web` or use `turbo dev` from root

For Users

No breaking changes - Website functionality remains the same:
- All pages load correctly
- All forms validate and display success messages
- Navigation and layout unchanged
- Design and styling preserved

Files Modified

Moved

- `/nextjs_space` → `/apps/web`
- `/components/navigation.tsx` → `/components/layout/navigation.tsx`
- `/components/footer.tsx` → `/components/layout/footer.tsx`

Created

- `/package.json` (root)

- `/turbo.json`
- `/packages/types/package.json`
- `/packages/types/src/index.ts`
- `/packages/types/tsconfig.json`
- `/packages/api-client/package.json`
- `/packages/api-client/src/index.ts`
- `/packages/api-client/src/ghl.ts`
- `/packages/api-client/src/validation.ts`
- `/packages/api-client/src/error-handling.ts`
- `/packages/api-client/tsconfig.json`
- `/README.md`
- `/PHASE_1_CHANGES.md` (this file)

Modified

- `/apps/web/app/layout.tsx` (import paths)
- `/apps/web/app/api/partnerships/route.ts` (removed Prisma, added validation)
- `/apps/web/app/api/contact/route.ts` (removed Prisma, added validation)
- `/apps/web/app/api/product-evaluation/route.ts` (removed Prisma, added validation)
- `/apps/web/app/api/joey-booking/route.ts` (removed Prisma, added validation)
- `/apps/web/lib/db.ts` (replaced with deprecation notice)
- `/apps/web/tsconfig.json` (added path mappings)

Deleted

- `/apps/web/prisma/schema.prisma`
- All database-related code

Testing

Verified

✅ TypeScript compilation passes with no errors
✅ Next.js build completes successfully
✅ All pages build without errors
✅ Dev server starts correctly
✅ Navigation works
✅ Forms render correctly
✅ Form validation works
✅ Turbo commands work (`turbo dev` , `turbo build`)

Not Yet Tested

⚠️ Form submission to GHL (Phase 2)
⚠️ Strapi integration (Phase 3)
⚠️ Production deployment (pending)

Next Steps

Phase 2: GoHighLevel Integration

1. **Get GHL credentials:**
   - API key
   - Location ID

2. **Implement GHL client:**
   - Complete `/packages/api-client/src/ghl.ts`
   - Add actual API calls
   - Map form fields to GHL contact fields

3. **Update API routes:**
   - Uncomment GHL integration code
   - Test all form submissions
   - Set up error monitoring

4. **Test workflows:**
   - Verify contacts created in GHL
   - Test automation triggers
   - Verify email notifications

Phase 3: Strapi CMS

1. **Initialize Strapi:**
   - Create `/apps/strapi`
   - Define content types
   - Set up Railway deployment

2. **Create API client:**
   - Add Strapi client to `/packages/api-client`
   - Add TypeScript types to `/packages/types`

3. **Build pages:**
   - Insights/Blog
   - Case Studies
   - Research Reports

Notes

- All existing functionality preserved
- No visual changes to the website
- Database removed but can be added back if needed
- Clean architecture ready for future growth
- Easy to add new apps (admin portal, mobile, etc.)

Questions?

Refer to the main README.md for:
- Getting started guide
- Development workflow

- Deployment strategy
- Technology stack details