Core Pages File Structure

Complete file reference for working on shared tenant pages

📂 Directory Overview

```
/home/ubuntu/healingbuds_website/nextjs_space/
├── app/
│   ├── page.tsx                      ← PLATFORM HOMEPAGE (budstack.to)
│   ├── about/
│   │   └── page.tsx                  ← Platform about page
│   ├── store/
│   │   └── [slug]/                   ← TENANT PAGES (you edit these)
│   │       ├── layout.tsx            ← Wraps all tenant pages with theme
│   │       ├── page.tsx              ← Homepage (template-specific)
│   │       ├── how-it-works/         ← ✏️ EDIT THIS
│   │       │   └── page.tsx
│   │       ├── conditions/           ← ✏️ EDIT THIS
│   │       │   └── page.tsx
│   │       ├── products/             ← ✏️ EDIT THIS
│   │       │   ├── page.tsx
│   │       │   └── [id]/
│   │       │       └── page.tsx
│   │       ├── consultation/         ← ✏️ EDIT THIS
│   │       │   ├── page.tsx
│   │       │   └── success/
│   │       │       └── page.tsx
│   │       ├── contact/              ← ✏️ EDIT THIS
│   │       │   └── page.tsx
│   │       ├── about/
│   │       │   └── page.tsx
│   │       ├── faq/
│   │       │   └── page.tsx
│   │       ├── privacy/
│   │       │   └── page.tsx
│   │       └── terms/
│   │           └── page.tsx
│   └── api/                          ← Backend endpoints
│       ├── consultation/
│       │   └── submit/
│       │       └── route.ts
│       └── store/
│           └── [slug]/
│               └── products/
│                   └── route.ts
├── components/
│   ├── consultation/                 ← ✏️ EDIT THESE (form components)
│   │   ├── consultation-form.tsx
│   │   └── steps/
│   │       ├── contact-step.tsx
│   │       ├── address-step.tsx
│   │       ├── conditions-step.tsx
│   │       ├── medical-history-part1-step.tsx
│   │       └── medical-history-part2-step.tsx
│   ├── ui/                           ← shadcn/ui components (reusable)
│   │   ├── button.tsx
│   │   ├── card.tsx
│   │   ├── input.tsx
│   │   ├── select.tsx
│   │   ├── alert.tsx
│   │   └── ...
│   ├── navigation.tsx                ← Global navigation (fallback)
│   ├── footer.tsx                    ← Global footer (fallback)
│   ├── tenant-theme-provider.tsx     ← Applies CSS variables
│   └── cart-dropdown.tsx             ← Shopping cart UI
├── lib/
│   ├── consultation-constants.ts     ← Hardcoded data (conditions, etc.)
│   ├── tenant.ts                     ← Tenant detection logic
│   ├── auth.ts                    ← Authentication utilities
│   └── utils.ts                   ← Shared utilities
├── public/                        ← Static assets
│   ├── core-pages/                ← ✏️ ADD YOUR ASSETS HERE
│   │   ├── conditions/
│   │   ├── how-it-works/
│   │   └── icons/
│   ├── templates/                 ← Template-specific assets
│   └── lovable-assets/            ← Shared template assets
└── templates/                     ← Template homepage designs
    ├── gta-cannabis/
    ├── wellness-nature/
    └── healing-buds-video/
```

📄 Page Files to Edit

0. Platform Homepage

File: `/app/page.tsx`

Current Lines: ~300 lines

Purpose: Market the BudStack SaaS platform to potential tenants

What to Keep:
- Hero section: “White-Label Medical Cannabis E-Commerce Platform”
- “Launch your own branded medical cannabis dispensary…”
- NFT license requirement messaging
- Key features showcase
- CTA to onboarding

What to Improve:
- Visual design and layout
- Feature presentation
- Trust indicators
- Mobile responsiveness

Key Dependencies:

```
import { Button } from '@/components/ui/button';
import Link from 'next/link';
// No tenant data - this is platform-level
```

Platform Data Available:

```
// Fetched from /api/super-admin/platform-settings
const settings = await getPlatformSettings();
// settings.businessName (e.g., "BudStack")
// settings.primaryColor
// settings.logoUrl
```

Important Notes:
- ❌ Does NOT use tenant CSS variables

- ❌ Does NOT adapt to tenant themes
- ✅ Uses fixed BudStack branding
- ✅ Links to `/onboarding` for signup

---

1. How It Works Page

File: `/app/store/[slug]/how-it-works/page.tsx`

Current Lines: ~200 lines

Purpose: Explain the 3-step consultation process

What to Keep:
- Step-by-step process explanation
- Trust indicators (certifications)
- FAQ section
- CTA to consultation page

What to Improve:
- Visual design of steps
- Icons/illustrations
- Mobile responsiveness
- Better CTA placement

Key Dependencies:

```
import { Tenant } from '@prisma/client';
import { getCurrentTenant } from '@/lib/tenant';
```

Tenant Data Available:

```
const tenant = await getCurrentTenant();
// tenant.businessName
// tenant.settings.primaryColor
// tenant.subdomain
```

---

2. Conditions Page

File: `/app/store/[slug]/conditions/page.tsx`

Current Lines: ~250 lines

Purpose: List medical conditions that can be treated

Hardcoded Data:

```
const conditions = [
  {
    name: 'Chronic Pain',
    icon: 'Zap',
    description: 'Persistent pain lasting more than 3 months...',
    symptoms: ['Persistent discomfort', 'Reduced mobility', ...],
    treatments: ['THC', 'CBD', 'Hybrid']
  },
  // ... more conditions
];
```

What to Keep:
- Grid layout
- Condition cards
- Icons for each condition
- Medical disclaimer

What to Improve:
- Filter by condition type
- Search functionality
- Better card design
- Modal with more details

Key Components:

```
import { Card } from '@/components/ui/card';
import { Badge } from '@/components/ui/badge';
import { Brain, Heart, Zap } from 'lucide-react';
```

---

3. Products Page

File: `/app/store/[slug]/products/page.tsx`

Current Lines: ~350 lines

Purpose: Display product catalog from Doctor Green API

Dynamic Data:

```
// Fetches from /api/store/[slug]/products
const products = await fetch(`/api/store/${slug}/products`);

// Product structure:
interface Product {
  id: string;
  name: string;
  price: number;
  thc_content: number;
  cbd_content: number;
  strain_type: 'indica' | 'sativa' | 'hybrid' | 'cbd';
  image_url: string;
  description: string;
}
```

What to Keep:
- Product grid
- Filter by strain type
- Sort options
- Add to cart functionality

What to Improve:
- Loading states
- Empty state design
- Filter UI
- Product card design
- Quick view modal

Key Components:

```
import { useCartStore } from '@/lib/cart-store';
import { Button } from '@/components/ui/button';
import { Select } from '@/components/ui/select';
import { Loader2 } from 'lucide-react';
```

---

4. Products Detail Page

File: `/app/store/[slug]/products/[id]/page.tsx`

Current Lines: ~400 lines

Purpose: Show detailed product information

Dynamic Data:

```
const product = await fetch(`/api/store/${slug}/products?id=${id}`);
```

What to Keep:
- Image gallery
- Product details (THC%, CBD%, strain)
- Effects list
- Medical uses
- Terpenes profile
- Add to cart

What to Improve:
- Image zoom functionality
- Related products section
- Reviews (future)
- Better quantity selector

---

5. Consultation Page

File: `/app/store/[slug]/consultation/page.tsx`

Current Lines: ~50 lines (wrapper only)

Purpose: Multi-step form for patient intake

Main Logic: Handled by `ConsultationForm` component

What to Keep:
- 5-step wizard
- Progress indicator
- Back/Next navigation
- Form validation
- Success page

What to Improve:
- Styling consistency
- Mobile UX
- Better validation messages
- Save progress (future)

Key Component:

```
import { ConsultationForm } from '@/components/consultation/consultation-form';
```

---

6. Contact Page

File: `/app/store/[slug]/contact/page.tsx`

Current Lines: ~200 lines

Purpose: Contact form and business information

Dynamic Data:

```
// From tenant settings
const contact = {
  email: tenant.settings?.contactEmail || 'info@example.com',
  phone: tenant.settings?.phone || '+44 20 1234 5678',
  address: tenant.settings?.address || 'London, UK'
};
```

What to Keep:
- Contact form
- Business info display
- Map integration (optional)
- Social links

What to Improve:
- Form design
- Success message
- Better layout
- Office hours display

🧩 Component Files to Edit

Consultation Form Components

Base Component: `/components/consultation/consultation-form.tsx`

Purpose: Main form wrapper with state management

Current Lines: ~400 lines

Key Features:
- Step navigation
- Form state management
- Progress tracking
- API submission

What to Improve:
- Better error handling
- Loading states
- Success animation

Step 1: `/components/consultation/steps/contact-step.tsx`

Fields:
- First name
- Last name
- Date of birth
- Email
- Phone
- Password

Validation:

```
const errors = {
  firstName: !data.firstName ? 'First name is required' : '',
  email: !isValidEmail(data.email) ? 'Valid email required' : '',
  // ...
};
```

Step 2: `/components/consultation/steps/address-step.tsx`

Fields:
- Shipping address (required)
- Business address (optional)
- Country/phone code

Combined: This step was merged from separate shipping/business steps

---

Step 3: `/components/consultation/steps/conditions-step.tsx`

Features:
- Checkboxes for conditions
- “Other” text input
- Validation (at least one required)

Hardcoded Data:

```
import { MEDICAL_CONDITIONS } from '@/lib/consultation-constants';
```

Steps 4 & 5: Medical History Parts 1 & 2

Files:
- `/components/consultation/steps/medical-history-part1-step.tsx`
- `/components/consultation/steps/medical-history-part2-step.tsx`

Fields: Various medical history questions (30+ fields total)

🔧 Utility Files to Know

Consultation Constants

File: `/lib/consultation-constants.ts`

Contains:

```
export const MEDICAL_CONDITIONS = [
  'Chronic Pain',
  'Anxiety',
  'Depression',
  // ...
];

export const COUNTRY_CODES = [
  { code: 'GB', name: 'United Kingdom', phoneCode: '+44' },
  { code: 'PT', name: 'Portugal', phoneCode: '+351' },
  // ...
];

export const BUSINESS_TYPES = [
  { value: 'sole_trader', label: 'Sole Trader' },
  { value: 'limited_company', label: 'Limited Company' },
  // ...
];
```

Usage: Import these for dropdown options, checkboxes, etc.

Tenant Utilities

File: `/lib/tenant.ts`

Key Functions:

```
// Get current tenant from request headers
export async function getCurrentTenant(): Promise<Tenant | null>

// Generate tenant URL
export function getTenantUrl(tenant: Tenant): string
```

Usage: Every page needs to call `getCurrentTenant()`

Cart Store

File: `/lib/cart-store.ts`

Purpose: Zustand store for shopping cart

Usage:

```
import { useCartStore } from '@/lib/cart-store';

const { items, addItem, removeItem, clearCart } = useCartStore();

addItem({
  id: product.id,
  name: product.name,
  price: product.price,
  quantity: 1,
  image: product.image_url
});
```

🎨 Component Library (shadcn/ui)

Available Components

Location: `/components/ui/`

Most Used:

```
import { Button } from '@/components/ui/button';
import { Card, CardHeader, CardTitle, CardDescription, CardContent } from '@/components/ui/card';
import { Input } from '@/components/ui/input';
import { Label } from '@/components/ui/label';
import { Select, SelectTrigger, SelectValue, SelectContent, SelectItem } from '@/components/ui/select';
import { Alert, AlertDescription } from '@/components/ui/alert';
import { Badge } from '@/components/ui/badge';
import { Skeleton } from '@/components/ui/skeleton';
import { Dialog, DialogTrigger, DialogContent, DialogHeader, DialogTitle } from '@/components/ui/dialog';
import { Tabs, TabsList, TabsTrigger, TabsContent } from '@/components/ui/tabs';
```

Usage Example:

```
<Card>
  <CardHeader>
    <CardTitle>Product Name</CardTitle>
    <CardDescription>Product description</CardDescription>
  </CardHeader>
  <CardContent>
    <Button>Add to Cart</Button>
  </CardContent>
</Card>
```

Note: These components already support theming via CSS variables!

🖼️ Asset Directories

Where to Put Your Assets

```
public/
├── core-pages/              ← YOUR NEW ASSETS GO HERE
│   ├── conditions/
│   │   ├── chronic-pain.jpg
│   │   ├── anxiety.jpg
│   │   ├── arthritis.jpg
│   │   └── ...
│   ├── how-it-works/
│   │   ├── step-1-book.svg
│   │   ├── step-2-consult.svg
│   │   ├── step-3-receive.svg
│   │   └── process-diagram.png
│   ├── products/
│   │   └── placeholder.jpg   ← Fallback for missing images
│   └── icons/
│       ├── medical.svg
│       ├── certification.svg
│       └── check-mark.svg
├── templates/               ← Template-specific (DON'T USE IN CORE PAGES)
│   ├── gta-cannabis/
│   ├── wellness-nature/
│   └── ...
└── lovable-assets/          ← Shared template assets (DON'T USE IN CORE PAGES)
```

Asset Naming Conventions

✅ **Good**:

```
chronic-pain-treatment.jpg
step-1-book-consultation.svg
certification-badge-mhra.png
```

❌ **Bad**:

```
image1.jpg
IMG_1234.PNG
untitled.svg
```

🔗 API Endpoints Reference

Products API

Endpoint: `/api/store/[slug]/products`

Method: GET

Query Params:
- `id` (optional): Get single product
- `category` (optional): Filter by category
- `strain` (optional): Filter by strain type

Response:

```json
{
  "products": [
    {
      "id": "prod_123",
      "name": "Blue Dream",
      "price": 45.00,
      "thc_content": 18.5,
      "cbd_content": 0.5,
      "strain_type": "hybrid",
      "image_url": "https://upload.wikimedia.org/wikipedia/commons/thumb/7/7d/American_medical_hashish%2810%29.jpg/250px-American_medical_hashish%2810%29.jpg",
      "description": "..."
    }
  ]
}
```

Consultation Submit API

Endpoint: `/api/consultation/submit`

Method: POST

Request Body:

```json
{
  "firstName": "John",
  "lastName": "Doe",
  "email": "john@example.com",
  "dateOfBirth": "1990-01-01",
  "shippingAddress": {
    "street": "123 Main St",
    "city": "London",
    "postcode": "SW1A 1AA",
    "country": "GB"
  },
  "medicalConditions": ["Chronic Pain", "Anxiety"],
  "medicalHistory": { /* 30+ fields */ }
}
```

Response:

```json
{
  "success": true,
  "clientId": "client_abc123",
  "message": "Consultation submitted successfully"
}
```

📦 Build & Deployment

Local Build

```bash
cd /home/ubuntu/healingbuds_website/nextjs_space
yarn build
```

Expected Output:

```
✓ Compiled successfully
✓ Generating static pages
✓ Finalizing page optimization

Route (app)                                   Size     First Load JS
┌ ○ /                                          190 B          102 kB
├ ƒ /store/[slug]                             34.1 kB         243 kB
├ ƒ /store/[slug]/how-it-works                1.44 kB         135 kB
├ ƒ /store/[slug]/conditions                  3.61 kB         108 kB
├ ƒ /store/[slug]/products                    4.69 kB         114 kB
└ ƒ /store/[slug]/consultation                164 B           160 kB

○  (Static)   prerendered as static content
ƒ  (Dynamic)  server-rendered on demand
```

Production Deployment

Automatic: When you push to main branch

Manual: Run from project root

```bash
cd /home/ubuntu/healingbuds_website
bash deploy.sh
```

🔍 Finding Things

Quick Search Commands

Find all pages:

```bash
find app/store -name "page.tsx" -type f
```

Find component usage:

```
grep -r "import.*Button" app/store/
```

Find CSS variable usage:

```
grep -r "var(--tenant-color" app/store/
```

Find hardcoded colors (to fix):

```
grep -r "bg-blue\|text-red\|border-green" app/store/
```

📚 Related Documentation

- **Theming Guide**: `COMPREHENSIVE_THEMING_GUIDE.md`
- **Designer Guide**: `CORE_PAGES_DESIGNER_GUIDE.md`
- **Local Setup**: `LOCAL_DEVELOPMENT_SETUP.md`
- **Architecture**: `MULTI_TENANT_ARCHITECTURE.md`
- **Template Design**: `TEMPLATE_DESIGN_GUIDE.md`

Last Updated: November 2025