

QuantumLeap AI Website - Developer Handover Document

Last Updated: November 9, 2025

Project: QuantumLeap AI Corporate Website

Repository: <https://github.com/AICyberSecurity911/WBSITE.git>

Live Site: <https://quantumleapai.abacusai.app>

Current Branch: `semi-final` (latest stable backup)

🎯 Project Overview

1. **Purpose:** Marketing website for QuantumLeap AI, targeting SMBs and Enterprise clients with AI automation, cybersecurity, and business transformation services.
 2. **Tech Stack:** Next.js 14.2.28, React 18.2, TypeScript 5.2.2, Tailwind CSS 3.3.3, Framer Motion 10.18.0, Prisma 6.7.0, PostgreSQL (Supabase).
 3. **Design System:** "Coral Reef" - Dark-mode-first palette with 8 accent colors (coral, cyan, purple, gold, lime, pink, teal, white).
 4. **Package Manager:** **ALWAYS use Yarn** (not npm/npx). Commands: `yarn add`, `yarn dev`, `yarn build`.
 5. **Project Structure:** Working directory is `/home/ubuntu/quantumleap_io/nextjs_space/` - all development happens here.
-

🎨 Design System: Coral Reef Palette

1. Primary Colors:

- Primary Background: `#121212` (secondary-bg)
- Primary Accent: `#FF7F50` (coral)
- Secondary Accent: `#40E0D0` (cyan/turquoise)
- Text Primary: White/high contrast
- Text Secondary: `text-gray-300`

2. 8 Accent Colors (defined in `globals.css` and `tailwind.config.ts`):

- Coral: `#FF7F50`
- Cyan: `#40E0D0`
- Purple: `#9370DB`
- Gold: `#FFC107`
- Lime: `#C6FF00`
- Pink: `#FF69B4`
- Teal: `#008080`
- White: `#FFFFFF`

3. Critical Design Rules:

- Light mode backgrounds use #f3f3f3 (grey), never pure white
- Buttons use coral (#ff5440) with gradient effects
- Headings in light mode use teal (#008080)
- Body text in light mode uses black (#111111)

4. Animation Guidelines:

All animations use Framer Motion with spring physics (stiffness: 200, damping: 35) for consistency.

Key Pages & Routes

1. Homepage (`/app/page.tsx`):

- Split-screen hero with video transition
- Trust bar with 14+ enterprise logos
- Value proposition (SMB vs Enterprise)
- Testimonials carousel (6-second auto-rotation)
- Links to service pages and SMB landing page

2. SMB Landing Page (`/app/smb/page.tsx`):

- High-converting sub-landing for small businesses
- Hero video (`/public/smb-hero-video.mp4`)
- 5 service cards (AI Workforce, Automation, Background Checks, Cyber Intelligence, Business Transformation)
- Single-card testimonials carousel with nav arrows
- “How It Works” 3-step section
- Trust bar with animated logos
- Theme toggle (light/dark mode)

3. About Us Page (`/app/about-us/page.tsx`):

- Founder’s vision (Paras Khurana - MIT/Caltech alumni)
- Elite cybersecurity credentials (NASA recognition)
- Strategic Advisory Board carousel (10 members, 2-card increments)
- Auto-rotating every 6 seconds with hover pause

4. Service Pages:

- `/app/ai-workforce/page.tsx` - AI team solutions
- `/app/intelligent-automation/page.tsx` - Process automation
- `/app/background-checks/page.tsx` - Beyond Background Checks™
- `/app/cyber-intelligence/page.tsx` - Threat intelligence
- `/app/business-transformation/page.tsx` - ROI calculator integration

5. Blog System (`/app/blog/[slug]/page.tsx`):

- Dynamic routing for blog posts
- Cybersecurity-themed CTAs
- Story-driven content structure
- SEO-optimized with schema markup

Technical Architecture

1. Database: PostgreSQL via Supabase

- Schema: `/database/schema.sql`
- Prisma ORM: `/prisma/schema.prisma`
- Connection config: `lib/supabase.ts`

2. Authentication: Admin authentication via JWT

- Implementation: `lib/admin-auth.ts`
- Admin dashboard: `/app/admin/` (protected routes)
- Login page: `/app/admin/login/`

3. Email System: Hostinger SMTP

- Config: `lib/email.ts` and `lib/email-automation.ts`
- Notification email: From credentials file
- Forms: Contact, consultation, lead magnets

4. Integrations:

- **TidyCal**: Appointment booking (`lib/tidycal.ts`)
- **Discord**: Lead notifications (`lib/discord.ts`, `lib/discord-automation.ts`)
- **Google Calendar**: Event sync (`lib/google-calendar.ts`)
- **Google Drive**: File storage (`lib/google-drive.ts`)
- **Firebase**: Admin features (`lib/firebase-admin.ts`)

5. API Routes (`/app/api/`):

- `/api/breach-check` - Cybersecurity checks
- `/api/calculator` - ROI calculations
- `/api/consultation` - Booking submissions
- `/api/contact-support` - Contact form
- `/api/lead-magnet` - Lead capture
- `/api/automation-lead` - Automation inquiries
- `/api/tidycal-webhook` - TidyCal event sync
- `/api/admin/*` - Admin dashboard APIs

Development Workflow

1. Starting Dev Server:

```
bash
cd /home/ubuntu/quantumleap_io/nextjs_space
yarn dev
```

- Runs on `http://localhost:3000`
- Hot reload enabled

2. Building for Production:

```
bash
cd /home/ubuntu/quantumleap_io/nextjs_space
yarn build
yarn start
```

3. **Testing:** No automated tests currently. Manual testing via browser.

4. Git Branches:

- `master` - Main production branch
- `Final-SMB` - Latest SMB page changes
- `semi-final` - Current stable backup (Nov 9, 2025)
- `SMB` - Previous SMB work

5. Deployment:

- **Platform:** Abacus.AI (Vercel-style)
 - **Domain:** quantumleapai.abacusai.app
 - **Process:** Push to GitHub → auto-deploys from designated branch
 - **Build command:** `yarn build`
-



Environment Variables & Credentials

1. Critical Files (not in Git):

- `/home/ubuntu/quantumleap_credentials.json` - All API keys/passwords
- `/home/ubuntu/quantumleap_io/nextjs_space/.env` - Environment variables
- Structure reference: `/home/ubuntu/.env.example`

2. Required Environment Variables:

- `DATABASE_URL` - Supabase PostgreSQL connection
- `SMTP_HOST`, `SMTP_PORT`, `SMTP_USER`, `SMTP_PASSWORD` - Email config
- `NOTIFICATION_EMAIL` - Support email address
- `DISCORD_WEBHOOK_URL` - Lead notifications
- `TIDYCAL_API_KEY` - Appointment booking
- `FIREBASE_*` - Admin authentication
- `GOOGLE_CALENDAR_*`, `GOOGLE_DRIVE_*` - Google integrations

3. Accessing Credentials:

```
bash
cat /home/ubuntu/quantumleap_credentials.json
```

- **NEVER commit this file to Git**
 - Contains sensitive API keys and passwords
-

Key Components & Sections

1. Hero Sections:

- `components/sections/coral-reef-hero.tsx` - Homepage split-screen hero with video
- `components/smb/hero-section.tsx` - SMB page hero with word-by-word animation
- Both use Framer Motion for scroll-triggered animations

2. Carousels:

- `components/sections/home-testimonials-carousel.tsx` - Homepage testimonials (6-second auto-rotation)
- `components/sections/advisory-board-section.tsx` - Advisory board (2-card increments, 6-second rotation)
- SMB testimonials - Built into `/app/smb/page.tsx` (single-card with arrows)

3. Calculators (`components/calculator/`):

- `ai-team-calculator.tsx` - AI workforce cost calculator
- `automation-calculator.tsx` - ROI for automation
- `hiring-risk-calculator.tsx` - Background check risk assessment
- `profit-potential-calculator.tsx` - Business transformation ROI

4. Exit Intent Modals:

- `components/automation-exit-intent.tsx` - Automation page lead magnet
- `components/background-checks-exit-intent.tsx` - Background checks guide
- `components/business-transformation-exit-intent.tsx` - Transformation checklist
- Trigger on mouse leave, scroll, or 30-second idle

5. Theme System:

- `components/smb/theme-context.tsx` - Theme provider with localStorage persistence
- `components/smb/theme-toggle.tsx` - Sun/moon switch button
- **Critical Fix:** Always provides context value even during initial mount (prevents hydration errors)

6. Layout Components:

- `components/layout/header.tsx` - Sticky nav with dark mode toggle, mobile menu
- `components/layout/footer.tsx` - Dark background (#121212), newsletter form, trust indicators

Common Issues & Fixes

1. Hydration Errors:

- **Cause:** `useTheme()` called before `ThemeProvider` wraps component
- **Fix:** Split component into two - outer wrapper uses `ThemeProvider`, inner component calls `useTheme`
- **Example:** See `/app/smb/page.tsx` structure (SMBPageContent wrapped by SMBPage)

2. Image Loading Issues:

- Always use Next.js `<Image>` component with `fill` prop inside fixed aspect ratio containers
- Add `relative` class to parent div
- Use meaningful alt text (not “image” or “picture”)

3. Build Failures:

- Run `yarn build` to check TypeScript errors
- Common issue: Missing dependencies → run `yarn install`
- Check `tsconfig.json` for strict mode issues

4. Duplicate Trust Bar Images:

- Intentional for smooth animation effects
- Creates seamless infinite loop in carousel

5. TidyCal Integration:

- API key stored in `/home/ubuntu/.config/abacusai_auth_secrets.json`
 - Access via: `json_data["tidycal"]["secrets"]["api_key"]["value"]`
 - Webhook endpoint: `/api/tidycal-webhook`
-

Performance & SEO

1. SEO Optimization:

- All pages have metadata files or inline metadata
- Schema markup for FAQs, testimonials, services
- OpenGraph images at 1200x630px
- Canonical URLs set for all pages

2. Performance:

- Lazy loading for images with `priority` flag for above-fold content
 - Code splitting via dynamic imports
 - Framer Motion animations use GPU-accelerated transforms
 - Video autoplay with `muted` attribute for mobile compatibility
-

Documentation Files

1. Key Documentation (in `/home/ubuntu/quantumleap_io/`):

- `CORAL_REEF DESIGN_SYSTEM.md` - Complete design specs
- `DESIGN_AUDIT_2024.md` - Design decisions and audit results
- `BREACH_CHECK_API_MIGRATION.md` - Cybersecurity API integration
- `AUTOMATION_SETUP_INSTRUCTIONS.md` - Email/Discord automation setup
- `DESIGN_IMPROVEMENTS_IMPLEMENTED.md` - Recent changes log

2. Asset References:

- Board member images: /public/board-members/*.png
 - Company logos: /public/company-logos/*.png
 - Videos: /public/hero-video.mp4 , /public/smb-hero-video.mp4
 - Founder photo: /public/founder-paras-khurana-new.jpg
-

Next Steps & TODO

1. Remaining Tasks:

- Implement blog CMS integration (currently static)
- Add remaining FAQ sections to SMB page
- Complete “Who This Is For” section on SMB page
- Integrate Google Analytics for conversion tracking
- Set up automated email sequences for lead nurturing

2. Testing Checklist:

- Test all forms (contact, consultation, lead magnets)
 - Verify email delivery for all automations
 - Test TidyCal webhook integration
 - Check mobile responsiveness on all pages
 - Test light/dark mode transitions on SMB page
-

Support & Resources

1. Repository Structure:

```
/home/ubuntu/quantumleap_io/nextjs_space/
    ├── app/                  # Next.js 14 app directory
    ├── components/           # Reusable React components
    ├── lib/                  # Utility functions, integrations
    ├── public/               # Static assets (images, videos)
    ├── prisma/              # Database schema
    ├── database/             # SQL files
    ├── hooks/                # Custom React hooks
    ├── types/                # TypeScript type definitions
    └── styles/               # Global CSS (globals.css)
```

2. Quick Commands Reference:

```
```bash
Install dependencies
yarn install
```

## Run dev server

---

yarn dev

## Build production

---

yarn build

## Generate Prisma client

---

yarn prisma generate

## Push to GitHub

---

```
git add -A
git commit -m "Your message"
git push origin [branch-name]
```

## Check current branch

---

git branch

## Switch branches

---

```
git checkout [branch-name]
...
```

## Key Contacts & Credentials Location

---

### 1. Credentials Access:

- Primary credentials: `/home/ubuntu/quantumleap_credentials.json`
- OAuth secrets: `/home/ubuntu/.config/abacusai_auth_secrets.json`
- Environment variables: `/home/ubuntu/quantumleap_io/nextjs_space/.env`

### 2. Admin Dashboard Access:

- URL: `https://quantumleapai.abacusai.app/admin/login`
- Credentials stored in database (check `quantumleap_credentials.json` for initial setup)

## Critical Reminders

---

1. **ALWAYS USE YARN:** Never use npm or npx. The project is configured for Yarn only.
  2. **NEVER COMMIT CREDENTIALS:** Ensure `.gitignore` includes `.env`, `quantum-leap_credentials.json`, and OAuth secrets.
  3. **TEST BEFORE PUSHING:** Run `yarn build` to catch TypeScript errors before pushing to GitHub.
  4. **MAINTAIN CORAL REEF PALETTE:** All new components must use the 8-accent color system defined in `globals.css`.
  5. **HYDRATION AWARENESS:** Always wrap components using `useTheme()` or browser APIs in proper context providers.
  6. **RESPONSIVE FIRST:** Test all changes on mobile, tablet, and desktop breakpoints.
- 

### **End of Handover Document**

For questions or clarifications, review the documentation files in `/home/ubuntu/quantumleap_io/` or examine recent Git commits on the `semi-final` branch.