

INFINTY FITNESS

Full Project Report - 2026 Rebuild

Website: <https://gym.putincode.com/> | Project: INFINTY FITNESS 2026 Professional CMS Rebuild

Executive summary

This rebuild upgrades the supplied gym equipment e-commerce project into a database-managed fitness platform for gym.putincode.com. It preserves the original catalogue, cart, orders, manual payment verification, user accounts and trainers while adding a professional modular CMS, page-level SEO, dynamic sitemap, visitor analytics with approximate location mapping, user access controls, improved security headers and a unified responsive visual system.

Existing database baseline

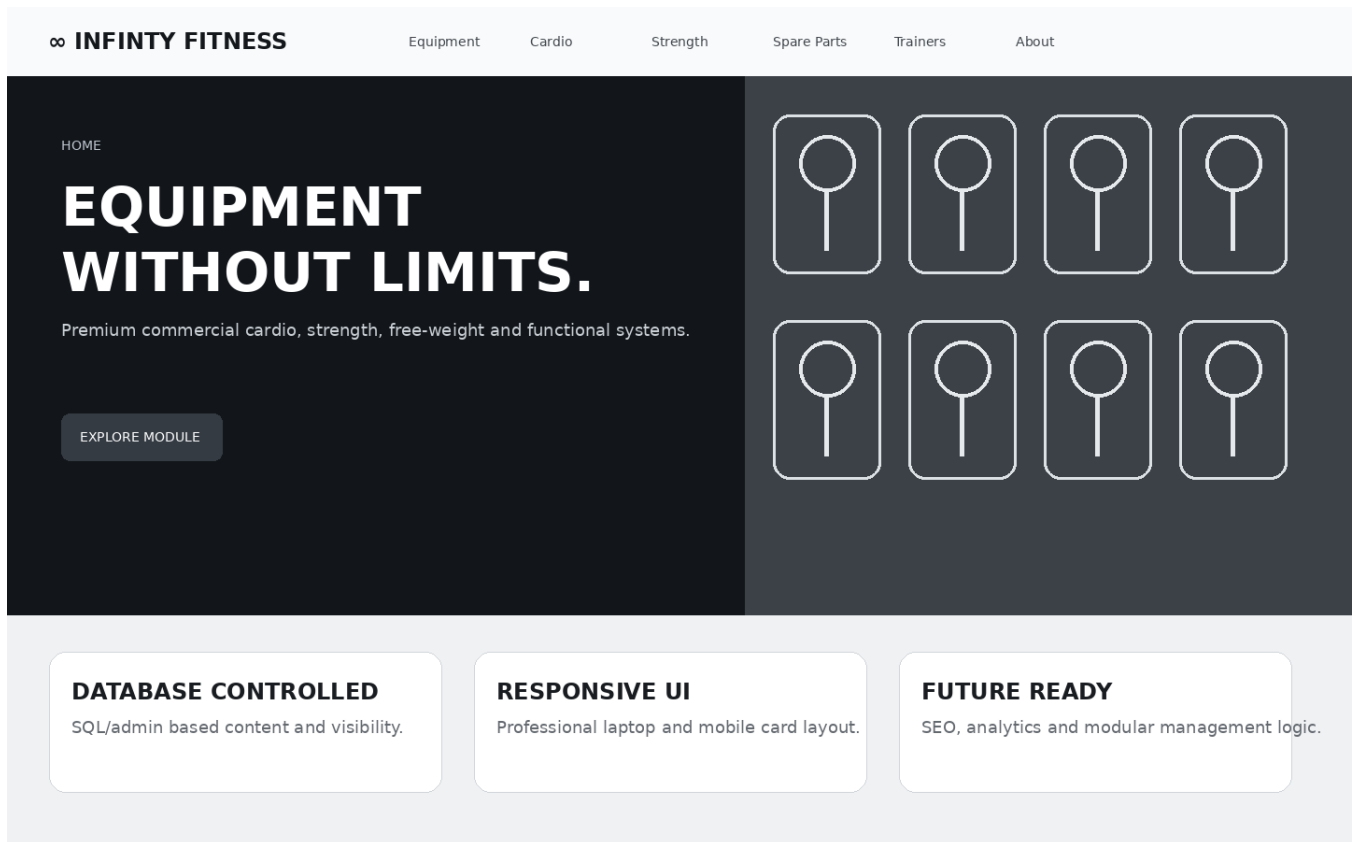
The supplied database already contained admins, categories, inquiries, orders, order_items, pages, payments, payment_methods, products, settings, trainers and users. The rebuild adds page_blocks, page_seo and visitor_logs instead of replacing those established business tables.

Architecture

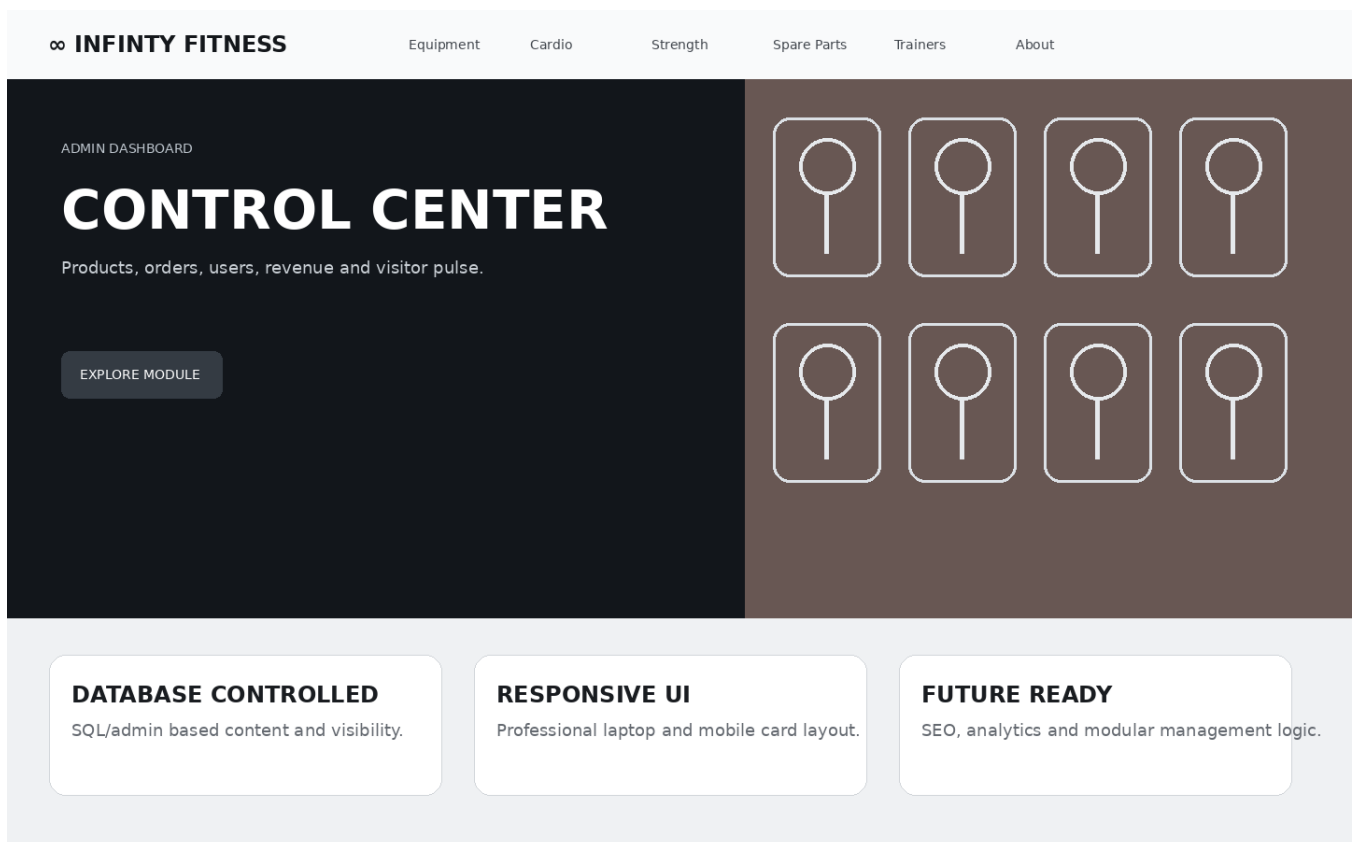
- config/config.php: application URL, database connection, session, CSRF, settings, upload and auth helpers.
- includes/functions.php: catalogue, cart, page, CMS block, SEO and rendering helpers.
- includes/header.php / footer.php: centralized metadata, OG tags, navigation, shared layout and analytics bootstrapping.
- admin/index.php: multi-module administrator control center.
- api/visit.php: resilient visitor event/geolocation capture.
- sitemap.php + .htaccess: dynamic sitemap endpoint and basic publication hardening.

Functional modules

Storefront modules: product/category browsing, live search, product details, cart, checkout, manual payment proof, customer registration/login/account, trainers, project enquiry and long-form company pages. Admin modules: dashboard, product/category/order/payment/user/trainer/enquiry management, page records, content blocks, SEO/sitemap, visitor analytics, payment methods, global settings and themes.



Public home design direction



Administrator dashboard direction

CMS design

The `page_blocks` table decouples long-form editorial content from PHP templates. Editors can create a page section with a small eyebrow heading, large title, long text, uploaded or external image, optional CTA, sort order and status. This supports future expansion without code changes and allows block-wise show/hide/delete control.

SEO design

`page_seo` stores the page slug, meta title, description, canonical URL, OG image, indexability and optional schema JSON. Header rendering automatically uses the record when available and falls back to global settings. The sitemap generator includes public routes and active product URLs.

Analytics design

`visitor_logs` records page-view events with session, IP, user agent, device category, available location fields, approximate coordinates, path, title, referrer and timestamp. The admin map uses Leaflet/OpenStreetMap. Geographic values are approximate and should be used for directional market analysis, not exact-person location.

Security posture

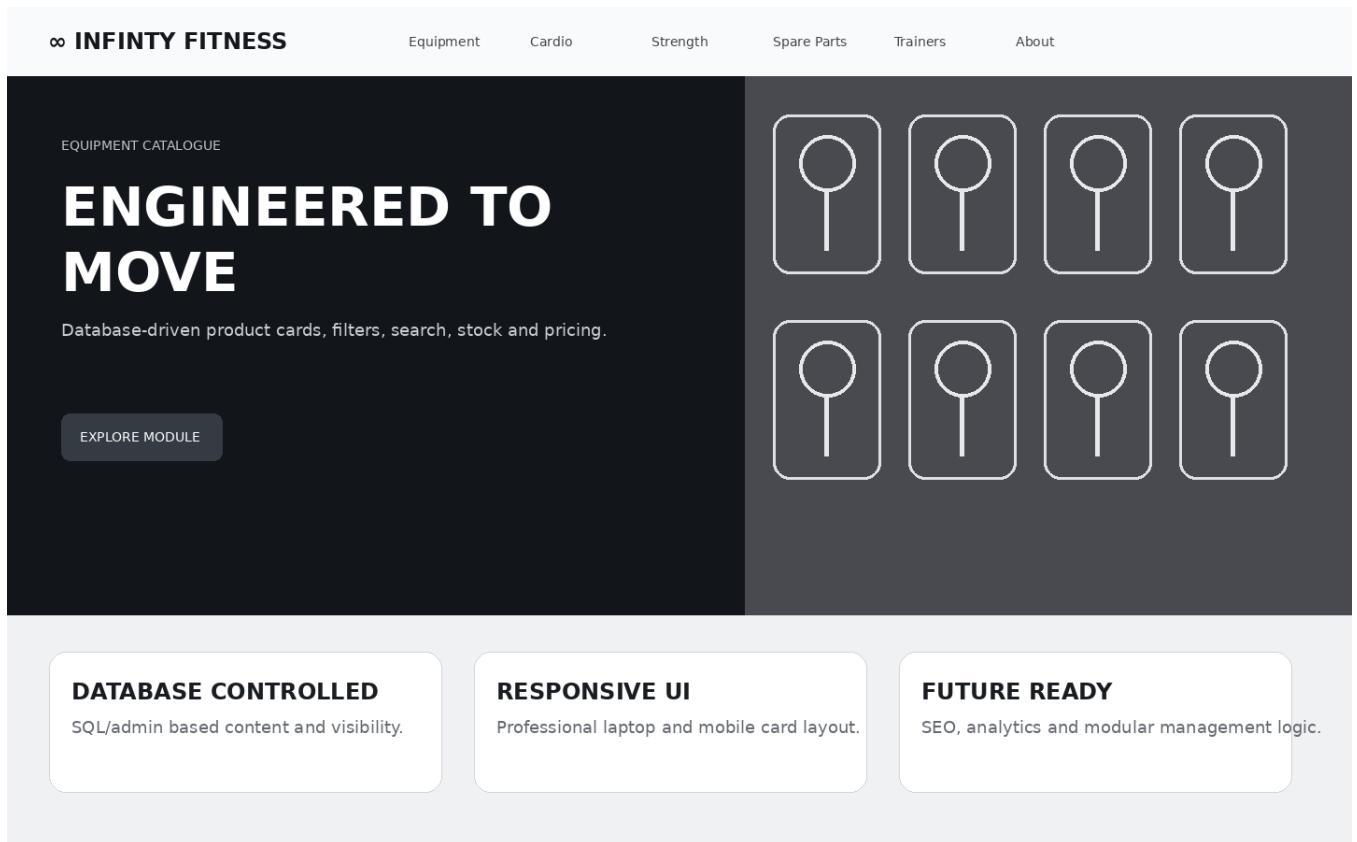
The project uses password hashing already present in the original account flow, CSRF validation on POST forms, admin/user status checks, HTTP security headers, upload extension allow-listing and SQL prepared statements for normal input paths. `.htaccess` disables directory listing and blocks SQL/README exposure. Production operators should rotate credentials, use HTTPS, back up the database and keep PHP current.

Validation completed

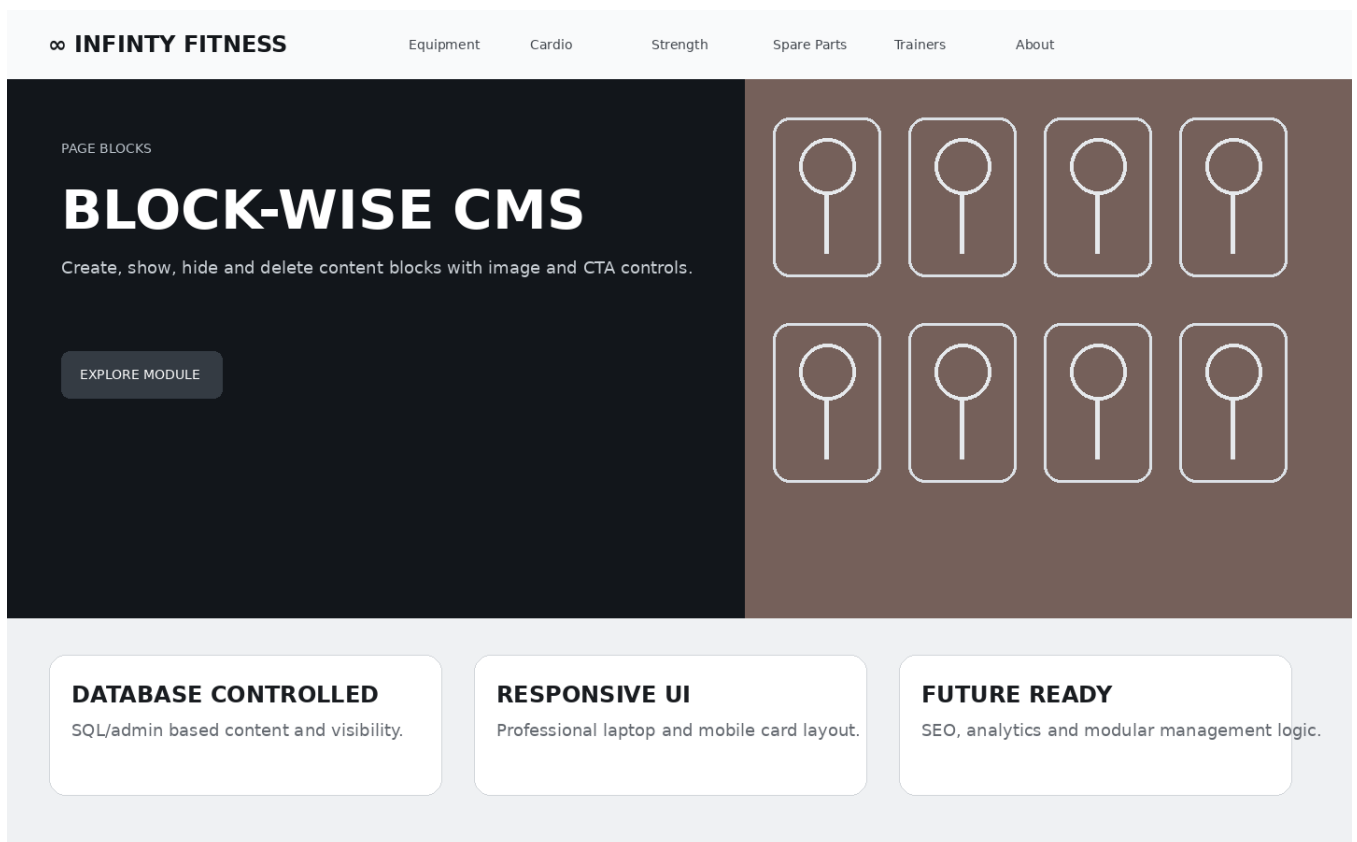
All PHP files in the rebuilt package were checked with the installed PHP CLI linter and returned no syntax errors. The database update is additive to existing business data. The package was prepared for the supplied MariaDB schema. Final production behavior still depends on the Hostinger PHP version, database permissions, outbound connectivity for external images/geolocation and domain DNS/SSL configuration.

Key files delivered

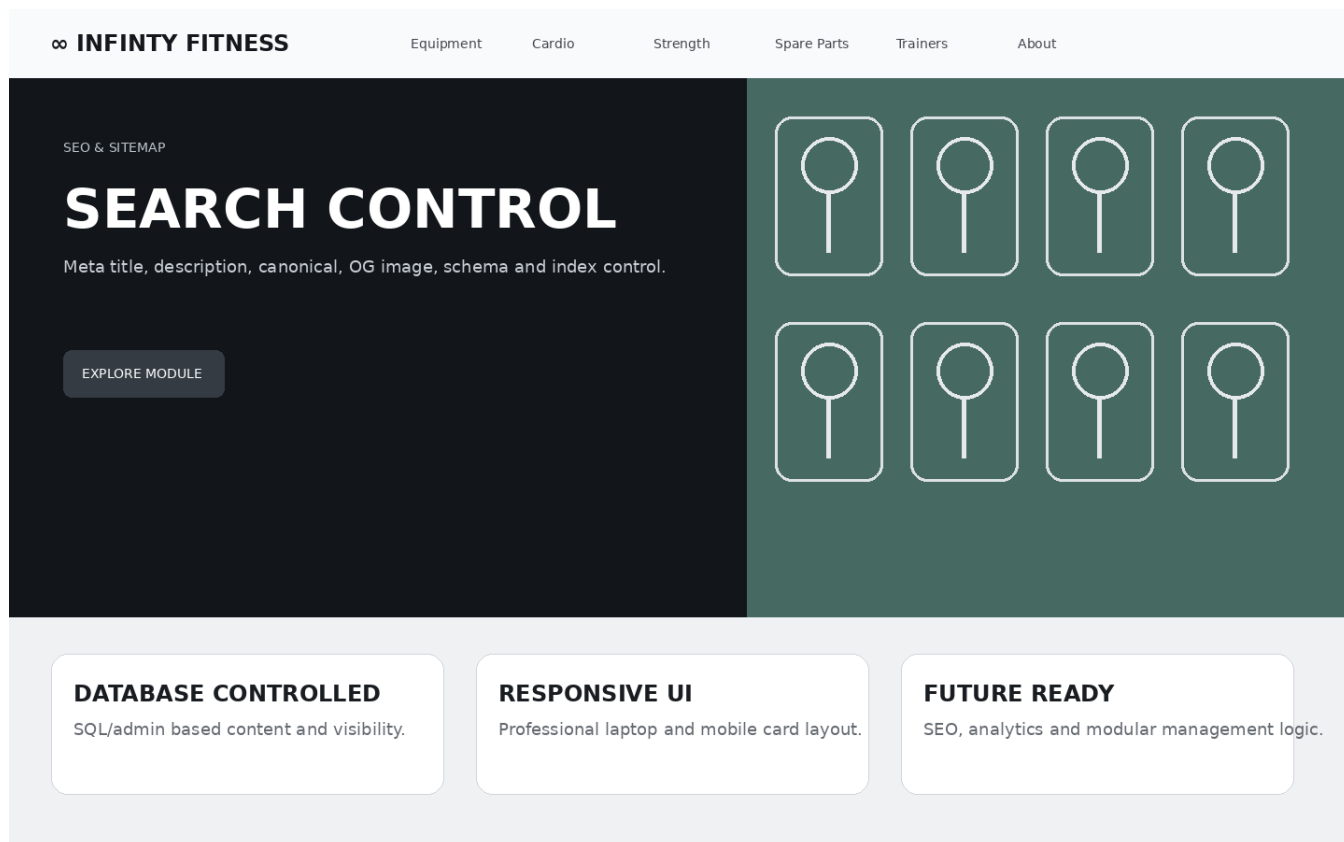
`install.sql` - fresh database install. `database_update_2026.sql` - upgrade existing database. `config/config.php` - production domain/database configuration. `admin/index.php` - administrator control center. `sitemap.php` - sitemap XML. `api/visit.php` - visitor analytics. Documentation PDFs - installation guide and project report.



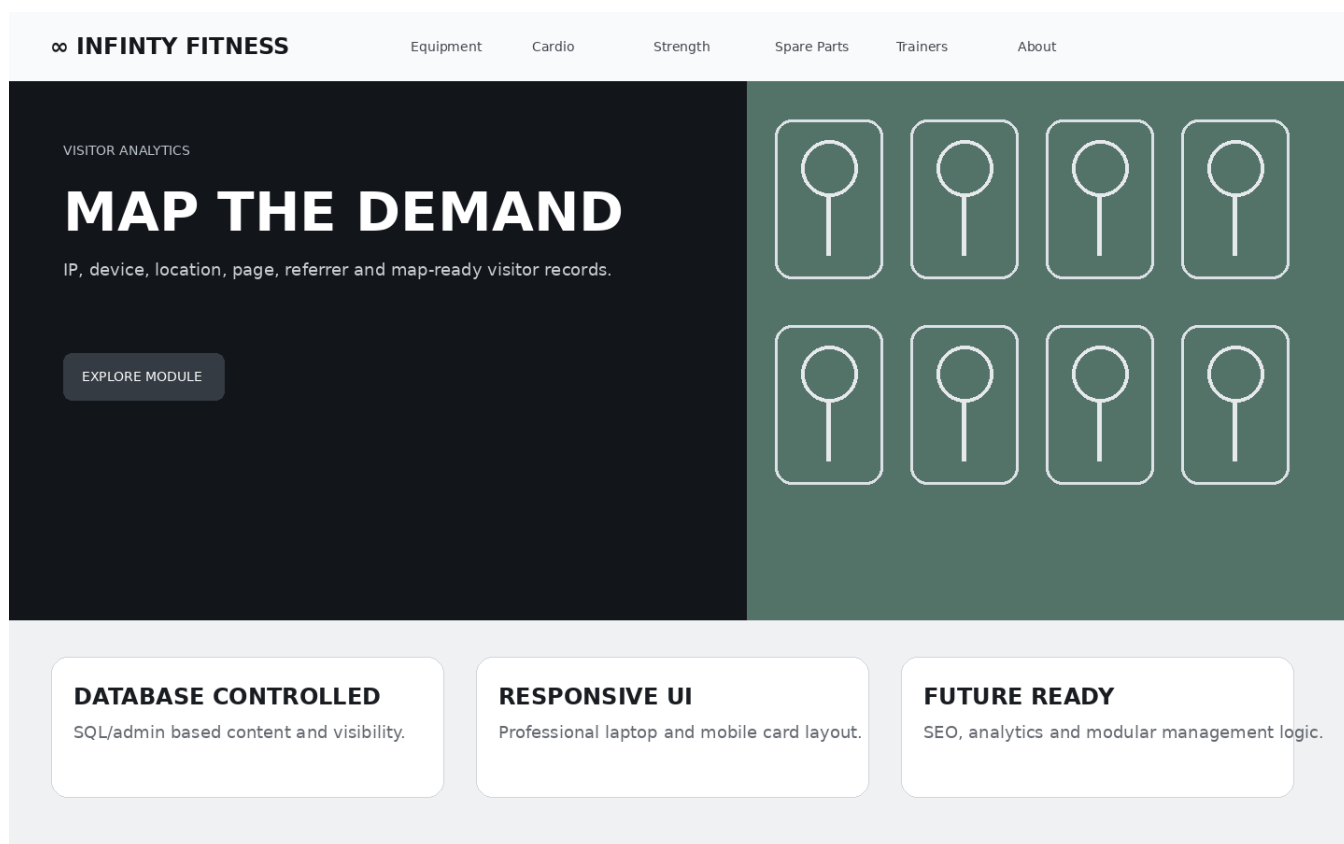
Catalogue



Page Blocks CMS



SEO controls



Visitor analytics

Deployment checklist

1. Upload project files to the gym.putincode.com document root.
2. Import the correct SQL file for fresh or existing installation.
3. Verify APP_URL and database credentials.
4. Set PHP 8.1+ where available and confirm uploads are writable.
5. Login to admin and change credentials/payment/contact details.
6. Review seeded page copy, prices, stock, images, payment methods, privacy policy and tracking setting.
7. Test registration, login, cart, checkout, enquiry, admin updates, sitemap and visitor records on the live domain.