Panduan Produksi MVP Roomah

Next.js 15 App Router + Supabase + Netlify

1. Ringkasan Arsitektur

Prinsip Modular Feature-Based

- Clean layering: Presentational (components) → Application (actions/queries) → Domain (schemas/validation) → Data (supabase)
- Feature isolation dengan shared components untuk reusability
- Server Actions untuk operasi tulis, RSC + TanStack Query untuk baca
- Cache invalidation via revalidateTag untuk real-time updates

Data Flow Pattern

- RSC untuk initial data loading (SEO-friendly)
- TanStack Query untuk client-side state management
- Server Actions untuk mutations dengan optimistic updates
- Real-time subscriptions untuk notifikasi dan riwayat taaruf

2. Struktur Folder & File

```
roomah/
 ├── .github/
 │   ├── workflows/
 │   │   ├── ci-cd.yml            # CI/CD pipeline dengan security scan
 │   │   ├── security-scan.yml
 │   │   └── lighthouse-audit.yml    # Performance monitoring
 │   └── ISSUE_TEMPLATE/
 ├── app/
 │   ├── (public)/
 │   │   ├── page.tsx             # Home (Guest dengan candidate listing)
 │   │   ├── tentang/page.tsx        # About Roomah company profile
│ │ └── layout.tsx          # Public layout dengan guest navigation
│ ├── (auth)/
│ │ ├── login/page.tsx         # Login existing users only
│ │ ├── register/page.tsx        # Register new users via Google
│ │ ├── onboarding/
│ │ │ ├── verifikasi/page.tsx    # 5Q verification dengan popup negatif
│ │ │ ├── cv/page.tsx          # CV wajib form onboarding
│ │ │ └── selesai/page.tsx       # Completion → redirect /cv-saya
│ │ └── layout.tsx          # Auth layout
│ ├── (app)/
│ │ ├── cari-jodoh/page.tsx       # Candidate search tanpa gender filter
│ │ ├── cv-saya/
│ │ │ ├── page.tsx           # CV preview/edit tabs
│ │ │ └── components/         # CV specific components
│ │ ├── riwayat-taaruf/
│ │ │ ├── page.tsx           # CV Masuk/Dikirim/Aktif tabs
│ │ │ └── components/
│ │ ├── koin-saya/
│ │ │ ├── page.tsx           # Balance & Midtrans sandbox
│ │ │ └── components/
│ │ └── layout.tsx          # App layout dengan user navigation
│ ├── (admin)/
│ │ ├── dashboard/page.tsx       # priority untuk MVP
│ │ └── layout.tsx
│ ├── api/
│   │   ├── health/route.ts          # System health monitoring
│   │   ├── midtrans/
│   │   │   └── webhook/route.ts     # Payment webhook dengan signature verification
│   │   ├── revalidate/route.ts      # Admin-protected cache invalidation
│   │   └── auth/
│   │       └── callback/route.ts    # Supabase auth callback
│   ├── globals.css                  # Tailwind + design tokens
│   ├── layout.tsx                   # Root layout
│   ├── loading.tsx                  # Global loading UI
│   ├── error.tsx                    # Global error UI dengan X-Request-ID
│   └── not-found.tsx
├── features/
│   ├── auth/
│   │   ├── components/              # Login/Register forms, onboarding steps
│   │   ├── server/
│   │   │   ├── actions.ts           # Registration, verification, session
│   │   │   └── queries.ts           # Auth queries
│   │   ├── schemas/                 # Zod validation untuk 5Q verifikasi
│   │   ├── store/                   # Auth state (Zustand)
│   │   └── types.ts
│   ├── candidates/
│   │   ├── components/
│   │   │   ├── candidate-card.tsx     # Kartu kandidat dengan avatar, kode, badge
│   │   │   ├── candidate-filters.tsx  # Filter dropdowns (gender auto untuk user)
│   │   │   ├── candidate-modal.tsx    # Profile detail modal
│ │ │ └── candidate-grid.tsx     # Grid dengan pagination prev/next
│ │ ├── server/
│ │ │ ├── actions.ts           # Ajukan taaruf dengan guards
│ │ │ └── queries.ts           # Query approved_candidates_v
│ │ ├── schemas/
│ │ ├── store/                 # Filter state
│ │ └── types.ts
│ ├── cv/
│ │ ├── components/
│ │ │ ├── cv-preview.tsx        # Preview tab (data wajib + avatar)
│ │ │ ├── cv-form.tsx           # Edit tab dengan 6 kategori
│ │ │ ├── biodata-section.tsx    # Biodata Lengkap
│ │ │ ├── ibadah-section.tsx     # Kondisi Ibadah
│ │ │ ├── kriteria-section.tsx   # Kriteria Pasangan
│ │ │ └── rencana-section.tsx    # Rencana Pernikahan
│ │ ├── server/
│ │ │ ├── actions.ts           # CV CRUD dengan normalisasi
│ │ │ └── queries.ts
│ │ ├── schemas/               # CV validation schemas
│ │ └── types.ts
│ ├── taaruf/
│ │ ├── components/
│ │ │ ├── cv-masuk-tab.tsx      # Pengajuan masuk dengan aksi
│ │ │ ├── cv-dikirim-tab.tsx    # Pengajuan terkirim dengan status
│ │ │ ├── taaruf-aktif-tab.tsx   # Taaruf aktif dengan kode
│   │   │   └── request-card.tsx      # Request item
│   │   ├── server/
│   │   │   ├── actions.ts            # Accept/reject dengan kode sequence
│   │   │   └── queries.ts
│   │   ├── schemas/
│   │   └── types.ts
│   ├── payments/
│   │   ├── components/
│   │   │   ├── koin-balance.tsx      # Saldo display
│   │   │   ├── topup-cards.tsx       # Top-up options sandbox
│   │   │   └── transaction-history.tsx
│   │   ├── server/
│   │   │   ├── actions.ts            # Create payment Midtrans
│   │   │   └── queries.ts            # Ledger balance queries
│   │   ├── schemas/
│   │   └── types.ts
│   ├── profile/
│   │   ├── components/
│   │   │   ├── avatar-upload.tsx     # Avatar management <1MB
│   │   │   └── profile-header.tsx
│   │   ├── server/
│   │   │   ├── actions.ts            # Profile updates
│   │   │   └── queries.ts
│   │   └── types.ts
│   ├── notifications/
│   │   ├── components/
│   │   │   └── notification-bell.tsx  # Placeholder untuk MVP
│   │   ├── server/
│   │   │   └── queries.ts
│   │   └── types.ts
│   └── admin/
│       ├── components/            # priority
│       ├── server/
│       └── types.ts
├── components/
│   ├── ui/                        # Base components
│   │   ├── button.tsx
│   │   ├── input.tsx
│   │   ├── dropdown.tsx
│   │   ├── modal.tsx
│   │   ├── tabs.tsx
│   │   ├── accordion.tsx
│   │   ├── badge.tsx
│   │   └── form/                  # Form components
│   ├── layout/
│   │   ├── header.tsx             # Navigation context-aware
│   │   ├── footer.tsx
│   │   ├── sidebar.tsx
│   │   └── mobile-nav.tsx         # Bottom tab bar user register
│   ├── common/
│   │   ├── loading-spinner.tsx
│   │   ├── error-boundary.tsx
│   │   ├── pagination.tsx
│   │   └── image-upload.tsx
│   └── providers/
│       ├── query-provider.tsx      # TanStack Query setup
│       ├── toast-provider.tsx
│       └── auth-provider.tsx
├── server/
│   ├── db/
│   │   ├── client.ts               # Supabase client config regional
│   │   └── types.ts                # Database types generated
│   ├── queries/
│   │   ├── candidates.ts           # approved_candidates_v queries
│   │   ├── cv.ts                   # CV queries dengan RLS
│   │   ├── taaruf.ts               # Taaruf queries
│   │   ├── payments.ts             # Ledger queries
│   │   ├── onboarding.ts           # 5Q verification queries
│   │   └── auth.ts                 # Auth queries
│   ├── services/
│   │   ├── midtrans.ts             # Payment service dengan signature
│   │   ├── sequence.ts             # Kode kandidat & taaruf generator
│   │   ├── email.ts                # Email service dengan deliverability
│   │   ├── image.ts                # Image processing thumbnails
│   │   ├── notification.ts         # Notification service
│ │  ├── audit-logger.ts         # Structured logging admin actions
│ │  ├── correlation.ts          # X-Request-ID management
│ │  ├── data-retention.ts       # Automated cleanup procedures
│ │  ├── credit-validator.ts     # Credit event validation
│ │  ├── webhook-hardening.ts    # Multi-layer rate limiting
│ │  ├── context-propagation.ts  # Async correlation handling
│ │  ├── debit-processor.ts      # Refund/chargeback handling
│ │  ├── webhook-throttling.ts   # Layered throttling implementation
│ │  ├── mv-metrics.ts           # MV performance tracking
│ │  ├── ledger-engine.ts        # Ledger-first operations
│ │  ├── payment-sequencer.ts    # Credit/debit sequencing
│ │  ├── canonical-correlation.ts # Server-side correlation
│ │  ├── financial-precision.ts  # Currency precision logic
│ │  ├── pii-safe-logger.ts      # Privacy-compliant logging
│ │  └── email-deliverability.ts # Bounce tracking mechanisms
│ ├── actions/
│ │  ├── candidates.ts           # Candidate actions
│ │  ├── cv.ts                   # CV CRUD actions
│ │  ├── taaruf.ts               # Taaruf actions dengan guards
│ │  ├── payments.ts             # Payment actions
│ │  ├── onboarding.ts           # Onboarding flow actions
│ │  └── auth.ts                 # Auth actions
│ └── revalidate/
│    ├── tags.ts                 # Cache tags constants
│    └── helpers.ts              # Revalidation helpers
```

```
├── lib/
│   ├── env/
│   │   ├── client.ts             # Client env vars
│   │   ├── server.ts             # Server env vars
│   │   └── validation.ts         # Env validation
│   ├── security/
│   │   ├── csrf.ts               # CSRF protection
│   │   ├── sanitize.ts           # Input sanitization dengan normalisasi
│   │   ├── rate-limit.ts         # Rate limiting implementation
│   │   └── headers.ts            # Security headers CSP/CORS
│   ├── analytics/
│   │   ├── ga4.ts                # Google Analytics
│   │   └── events.ts             # Custom events
│   ├── logger/
│   │   ├── client.ts             # Client logging
│   │   └── server.ts             # Server logging dengan correlation
│   ├── validators/
│   │   ├── auth.ts               # Auth schemas
│   │   ├── cv.ts                 # CV schemas 6 kategori
│   │   ├── taaruf.ts             # Taaruf schemas
│   │   └── common.ts             # Common schemas
│   ├── utils/
│   │   ├── date.ts               # Date utilities timezone
│   │   ├── text.ts               # Text formatting Capitalize Each Word
│   │   ├── currency.ts           # Currency formatting IDR
│   │   └── constants.ts          # App constants
│   └── images/
│       ├── upload.ts             # Image upload validation
│       ├── resize.ts             # Thumbnail processing
│       ├── optimize.ts           # Image optimization WebP/AVIF
│       └── remote-patterns.md    # CDN domain patterns
├── stores/
│   ├── query-keys.ts             # TanStack Query keys
│   ├── filters.ts                # Filter state (Zustand)
│   ├── ui.ts                     # UI state modals
│   └── auth.ts                   # Auth state
├── styles/
│   ├── tokens.css                # Design tokens HSL
│   └── globals.css               # Global styles Tailwind
├── public/
│   ├── icons/
│   │   └── roomah-logo.svg
│   ├── images/
│   │   ├── hero-banner.webp
│   │   ├── default-avatar-male.webp   # <50KB default
│   │   ├── default-avatar-female.webp  # <50KB default
│   │   └── placeholder-banner.webp
│   ├── manifest.json             # PWA manifest
│   ├── robots.txt
│   └── sitemap.xml
```

```
├── supabase/
│   ├── migrations/
│   │   ├── 20241201_initial_schema.sql
│   │   ├── 20241202_rls_policies.sql
│   │   └── 20241203_indexes.sql
│   ├── policies/
│   │   ├── profiles.sql            # RLS policies
│   │   ├── cv.sql
│   │   ├── approved_candidates_v.sql   # View publik RLS
│   │   ├── onboarding_verifications.sql # Owner-only RLS
│   │   └── transactions.sql
│   ├── seeds/
│   │   ├── test-users.sql
│   │   └── sample-data.sql
│   └── edge-functions/
│       └── check-user-exists/        # Google OAuth user check
│           ├── index.ts
│           └── README.md
├── docs/
│   ├── adr/                      # Architecture decisions
│   │   ├── 001-tech-stack.md
│   │   ├── 002-auth-strategy.md
│   │   └── 003-payment-flow.md
│   ├── api/
│   │   ├── contracts.md            # API contracts
│ │ └── webhook-specs.md        # Midtrans payload, retry, idempotency
│ ├── security/
│ │ ├── baseline.md            # Security requirements
│ │ ├── step-up-auth.md         # Admin re-auth scope & UX
│ │ ├── anti-enumeration.md      # Generic messaging patterns
│ │ ├── webhook-hardening.md      # Rate limits, replay protection
│ │ ├── webhook-throttling-layers.md # Tunable baselines + backpressure
│ │ ├── replay-vs-idempotency.md   # Clock drift tolerance
│ │ ├── unlock-deliverability.md   # Email delivery & anti-enumeration
│ │ └── threat-model.md
│ ├── runbook/
│ │ ├── deployment.md
│ │ ├── monitoring.md
│ │ ├── rollback.md
│ │ ├── auth-google-nonce.md      # OAuth troubleshooting
│ │ └── sequences.md            # Sequence rules, reset, rollback
│ ├── db/
│ │ ├── materialized-views.md     # Refresh strategy, monitoring
│ │ └── mv-refresh-metrics.md     # Performance metrics & strategies
│ ├── perf/
│ │ └── image-guidelines.md      # Target size per context
│ ├── ops/
│ │ ├── webhook-retry.md         # Retry sequence, backoff, audit
│ │ ├── scheduled-jobs-fallback.md  # Manual procedures
│ │ └── slo-monitoring.md        # Service level objectives
│   ├── compliance/
│   │   ├── data-retention.md        # Retention rules, cleanup
│   │   ├── field-exposure.md        # Public/private field matrix
│   │   └── privacy-policy.md        # User rights, export/deletion
│   ├── cost/
│   │   ├── netlify-credits.md       # Usage monitoring, 70%/90% alarms
│   │   └── monitoring-sources.md    # Metric sources & fallback
│   ├── forensics/
│   │   ├── payment-audit.md         # Audit trail structure
│   │   ├── payment-event-matrix.md  # Credit events per payment_type
│   │   └── refund-chargeback.md     # Status FINAL mapping per kanal
│   ├── wallet/
│   │   └── ledger-model.md          # Ledger-first & financial precision
│   ├── observability/
│   │   ├── request-context.md       # Async correlation propagation
│   │   ├── id-correlation-tooling.md  # Sentry/logs/client integration
│   │   └── canonical-request-id.md  # Server-side ID & PII-safe logging
│   └── security/
│       └── progressive-auth.md      # CAPTCHA & lockout policies
├── tests/
│   ├── unit/
│   │   ├── components/
│   │   ├── utils/
│   │   └── validators/
│   ├── integration/
│   │   ├── auth.test.ts             # OAuth flow, onboarding
│   │   ├── taaruf-flow.test.ts       # Complete taaruf process
│   │   └── payment.test.ts           # Midtrans webhook, idempotency
│   ├── e2e/
│   │   ├── guest-flow.spec.ts        # Home → ajukan → redirect
│   │   ├── onboarding.spec.ts        # 5Q → CV → selesai
│   │   ├── cv-management.spec.ts     # CV lifecycle
│   │   └── taaruf-process.spec.ts    # End-to-end taaruf
│   ├── fixtures/
│   │   ├── users.json
│   │   └── cv-data.json
│   └── setup.ts
├── scripts/
│   ├── seed-database.ts             # Database seeding
│   ├── backup-restore.ts            # Backup utilities
│   ├── rotate-keys.ts               # Key rotation
│   └── cleanup-expired.ts           # Data cleanup
├── config/
│   ├── csp.ts                       # Content Security Policy
│   ├── cors.ts                      # CORS configuration
│   └── headers.ts                   # Security headers
├── middleware.ts                    # Route protection & correlation
├── next.config.ts                   # Next.js configuration
├── .env.example                     # Environment template
├── .env.local                       # Local environment
```

```
 ├── tailwind.config.ts          # Tailwind configuration
 ├── tsconfig.json             # TypeScript config
 ├── package.json
 └── README.md
```

3. Mapping Fitur ↔ Folder

Auth Feature

- **Components**: Login/register forms, onboarding 3 steps (verifikasi 5Q, CV wajib, selesai)
- **Server**: Registration dengan Google OAuth, verificasi kesiapan, session management
- **Schemas**: Input validation untuk 5Q verification dengan popup handling negatif
- **Kontrak**: User registration → onboarding flow → redirect /cv-saya

Candidates Feature

- **Components**: Card grid dengan avatar/kode/badge, filters dropdown, profile modal, pagination
- **Server**: Search queries approved_candidates_v, ajukan taaruf dengan business guards
- **Store**: Filter state dengan auto gender filter untuk user register
- **Kontrak**: Home/Cari Jodoh → filtered candidates dengan auto gender → profile modal → ajukan taaruf

CV Feature

- **Components**: Preview/edit tabs, 6 kategori sections (biodata lengkap/kondisi fisik/latar belakang keluarga/ibadah/kriteria/rencana)
- **Server**: CV CRUD dengan normalisasi Capitalize Each Word, admin verification
- **Schemas**: Validation untuk semua kategori CV lengkap
- **Kontrak**: Onboarding CV → CV Saya → admin approval → candidate visibility dengan kode

Taaruf Feature

- **Components**: 3 tabs (masuk/dikirim/aktif), request cards dengan aksi
- **Server**: Accept/reject actions, status updates, kode sequence generation
- **Kontrak**: Ajukan taaruf → CV Dikirim → response → CV Masuk → Taaruf Aktif dengan kode

Payments Feature

- **Components**: Balance display, top-up cards sandbox, transaction history
- **Server**: Midtrans integration, ledger-first balance updates
- **Kontrak**: Insufficient koin → Koin Saya → top-up Midtrans → ajukan taaruf enabled (koin sesuai konfigurasi, baseline tunable)

4. App Router & Guards

Route Groups

- **(public)**: Home, Tentang - accessible semua dengan candidate browsing
- **(auth)**: Login (existing users), Register (new via Google), Onboarding - redirect jika authenticated
- **(app)**: Main app features - require authentication dengan CV/koin/taaruf guards
- **(admin)**: Admin features - require admin role email-only (non-priority)

Middleware Guards

- Authentication check untuk route groups
- CV status validation untuk ajukan taaruf (approved required)
- Koin balance check untuk payment actions (koin sesuai konfigurasi, baseline tunable)
- Active taaruf check untuk new requests (tidak boleh multiple)
- Admin provider validation (email-only, Google OAuth ditolak)

Redirect Logic

- Guest → Ajukan Taaruf → Login/Register
- CV Status Review/Revisi → CV Saya Edit tab
- Insufficient koin → Koin Saya

- Onboarding incomplete → Onboarding steps
- Onboarding selesai → /cv-saya (default landing)

5. Auth & OAuth

Google OAuth Implementation

- **Nonce Handling**: GIS kirim SHA-256(base64url) ke Google, raw nonce ke Supabase signInWithIdToken
- **Skip Nonce Checks**: OFF di production untuk security compliance
- **Edge Function**: check-user-exists WAJIB untuk membedakan login vs register flow
- **Admin Policy**: WAJIB provider=email, Google OAuth tidak diizinkan untuk admin

Security Controls

- **Step-up Re-auth**: Aksi admin sensitif memerlukan re-authentication <15 menit
- **Progressive Cooldown**: 1st fail = 5min, 2nd = 15min, 3rd = 60min, reset 24h success
- **Email Unlock**: Single-use token valid 1 jam, limit 3 unlock per email per 24h
- **Anti-enumeration**: Generic messaging tanpa expose account existence

6. Database & RLS

Entitas Utama

- **profiles**: User basic info + verification status
- **cv_data**: Detailed CV dengan 6 kategori (biodata/ibadah/kriteria/rencana)
- **approved_candidates_v**: View untuk public listing (single source of truth)
- **onboarding_verifications**: Data 5Q dengan RLS owner-only
- **taaruf_requests**: Pengajuan taaruf dengan status tracking
- **taaruf_active**: Active taaruf dengan kode sequence
- **ledger_entries**: CREDIT/DEBIT sebagai single source of truth
- **wallet**: Balance calculation dari ledger aggregation

RLS Policies

- **Deny-by-default**: Semua tabel dengan explicit permissions

- **Public View**: approved_candidates_v hanya SELECT untuk listing
- **Owner-only**: CV data, taaruf requests, onboarding verifications

Index Strategy

- **Composite Index**: approved_candidates_v filtering (gender, age, province, education) pada tabel sumber
- **Sequence Generation**: index untuk last_sequence, created_at
- **Active Taaruf**: partial index untuk guard checking
- **Ledger Performance**: index user_id, created_at, transaction_type

Materialized Views

- **Refresh Strategy**: Event-driven via enqueue + scheduler (pg_cron/Netlify Functions)
- **Prerequisites**: REFRESH CONCURRENTLY memerlukan unique index valid
- **Fallback**: Non-concurrent refresh di off-peak window <30 detik SLA
- **Size-aware**: Atomic view swap >500MB, source partitioning >1GB
- **Metrics**: Track avg/p95 refresh duration, lock contention

7. State & Data Fetching

TanStack Query Strategy

- **Query Keys**: candidates (filter-based), cv-preview, taaruf-history, koin-balance
- **Cache Strategy**: SSG untuk Home (ISR 1 hour), RSC untuk initial loading
- **Deduplication**: 2 menit interval, revalidateOnFocus false
- **Real-time**: Supabase subscriptions untuk notifications

Zustand State Management

- **Filter State**: Candidate search filters dengan gender auto-handling
- **UI State**: Modal visibility, loading states
- **Auth State**: User session dengan role management

Revalidation Tags

- **candidates-list**: Invalidate saat CV approved

- **user-balance**: Invalidate saat payment success
- **taaruf-requests**: Invalidate saat status change
- **transaction-history**: Invalidate setelah ledger update

8. API & Payments (Midtrans)

Payment Lifecycle Management

- **Credit Event**: Sekali per order_id pada settlement/capture status FINAL
- **Debit Event**: Refund/void/chargeback dengan mapping per payment_type

Payment Type Specific Mapping

- **Status Matrix**: VA (expire/cancel), CC (refund/chargeback), QRIS (chargeback), e-wallet (refund/reversal)
- **Reference Table**: Mapping status FINAL disimpan di docs/forensics/refund-chargeback.md dengan URL dokumentasi resmi + version tracking per kanal/acquirer
- **Sequence Validation**: DEBIT tanpa prior CREDIT = no-op + audit trail

Idempotency Implementation

- **Deterministic Keys**: order_id + "CREDIT" atau order_id + "DEBIT"
- **Deduplication Table**: Unique constraint pada idempotency_key
- **Race Protection**: Atomic operations untuk concurrent callbacks
- **Audit Integration**: Correlation antara business events dan ledger entries

Webhook Security & Resilience

- **Signature Verification**: SHA512(order_id + status_code + gross_amount + serverKey)
- **Multi-layer Throttling**: Global (1000/min), per-IP (100/min), per-order (10/min) - baseline tunable
- **Backpressure Queue**: Circuit breaker untuk burst handling
- **Forensic Headers**: User-Agent hash, IP hash, timestamp untuk investigation
- **Replay Protection**: ±5 menit window dengan late replay flagging

Financial Precision

- **Ledger-first Model**: Balance = SUM(ledger_entries), no direct mutations
- **Currency Handling**: IDR sen precision dengan consistent rounding
- **Amount Validation**: Gross amount vs original order dengan fee/discount reconciliation
- **Freeze Mechanism**: Wallet state change tanpa balance modification

9. Performance & Images

Rendering Strategy

- **SSG**: Home, Tentang (SEO critical pages)
- **ISR**: Candidate listing dengan frequent updates
- **SSR**: Personal pages (CV Saya, Riwayat Taaruf)
- **CSR**: Interactive components dengan optimistic updates

Image Optimization

- **Avatar Delivery**: Thumbnails 20-100KB via Next/Image + CDN, bukan file sumber
- **Default Assets**: Public bucket dengan compressed WebP/AVIF <50KB
- **Private Bucket**: User avatars dengan signed URLs, <1MB upload limit
- **Lazy Loading**: Non-critical images dengan intersection observer

CDN & Regional Performance

- **Functions Region**: ap-southeast-1 untuk proximity Supabase & Midtrans
- **CDN Global**: Netlify/Vercel Edge dengan cache optimization
- **Connection Pooling**: Sesuai free tier limits untuk database efficiency

10. Security Implementation

Content Security & Headers

- **CSP**: Strict policy dengan nonce-based inline scripts
- **CORS**: Configured untuk webhook domains dan API endpoints
- **Security Headers**: HSTS, X-Content-Type-Options, X-Frame-Options, Referrer-Policy

Input Validation & Sanitization

- **Zod Schemas**: Semua forms dengan comprehensive validation
- **Text Normalization**: Capitalize Each Word untuk CV inputs
- **File Upload**: Type, size validation dengan security scanning
- **XSS Prevention**: Content sanitization dengan DOMPurify equivalent

Rate Limiting Strategy

- **Login/Register**: 5 attempts/15min per IP, progressive CAPTCHA after 3 fails
- **Ajukan Taaruf**: 3 requests/hour per user dengan business guard validation
- **Server Actions**: 30 requests/min per authenticated user
- **Admin Actions**: 300 requests/min (10x user) + step-up re-auth untuk sensitive operations

11. Observability & Monitoring

Request Correlation

- **Canonical X-Request-ID**: Server-generated untuk semua operations
- **Context Propagation**: Across Server Actions, RSC, queue workers, webhook processing
- **Response Headers**: X-Request-ID dalam success dan error responses
- **PII-safe Logging**: Hanya correlation IDs, status codes, reason codes

Error Response Standard

- **Format**: Error code + generic message + X-Request-ID tanpa PII exposure
- **Critical Operations**: Payment/webhook/auth/taaruf state-changing actions
- **Support Integration**: Correlation ID untuk customer issue escalation

Monitoring Stack

- **Error Tracking**: Sentry dengan correlation-id tags
- **Structured Logging**: JSON format dengan correlation, user_id, action metadata
- **Performance Metrics**: Core Web Vitals, database query performance
- **Business Metrics**: Registration funnel, taaruf success rates, payment conversion

Audit & Compliance

- **Admin Actions**: Approve/reject CV, manual top-up, role changes dengan correlation-id
- **Payment Trail**: Complete linking antara business events dan ledger entries
- **Data Retention**: 2 tahun CV/taaruf, 5 tahun transaction logs
- **Privacy Controls**: Field-level exposure matrix, cookie consent, export/delete workflow

12. Operations & Cost Management

Regional Configuration

- **Supabase Functions**: ap-southeast-1 (Singapore) untuk latency optimization
- **Netlify Edge**: Regional alignment dengan database proximity
- **Connection Management**: Pooling sesuai free tier limits

Cost Monitoring & Alarms

- **Primary Sources**: Netlify Credits API, Supabase Usage API, CDN analytics
- **Sampling Frequency**: Real-time untuk critical, hourly untuk trends
- **Alert Thresholds**: 70% early warning, 90% critical action
- **Fallback Sources**: Manual exports, log scraping, ETL processes saat API down
- **Response Procedures**: Cache optimization, deploy deferral, feature flag disable

Scheduled Jobs & Reliability

- **Job Types**: DLQ processing, MV refresh, data retention cleanup
- **Execution**: pg_cron atau Netlify Scheduled Functions
- **SLA**: DLQ drain ≤1 hour, daily sweep sebelum 03:00 WIB
- **Fallback**: Manual procedures documented dengan clear escalation

13. Compliance & Privacy

Data Protection

- **Field Exposure**: Public (age, education, province) vs Private (address, salary detail)
- **PII-safe Logging**: Exclude CV data dan personal identifiers

- **Cookie Consent**: Necessary, analytics, marketing categories
- **User Rights**: Data export, deletion request workflow

Retention Policies

- **CV/Taaruf Data**: 2 tahun setelah inactive
- **Transaction Logs**: 5 tahun untuk compliance
- **Session Data**: Standard TTL dengan cleanup procedures
- **Audit Trails**: Permanent retention untuk regulatory compliance

14. Testing Strategy

Unit Testing

- **Components**: CV forms, candidate cards, payment components
- **Utilities**: Text normalization, currency formatting, date handling
- **Validators**: Zod schemas untuk auth, CV, taaruf, payment

Integration Testing

- **Auth Flow**: OAuth nonce, onboarding 5Q, progressive cooldown
- **Payment Cycle**: Midtrans webhook, idempotency, credit/debit lifecycle
- **Taaruf Process**: End-to-end dari ajukan sampai active dengan kode

E2E Testing

- **Guest Flow**: Home browsing → ajukan taaruf → redirect login/register
- **User Journey**: Onboarding → CV management → taaruf process → payment
- **Admin Actions**: CV approval, manual transactions dengan audit verification
- **Error Scenarios**: Payment failures, webhook retries, rate limiting

15. Acceptance Checklist Komprehensif

Authentication & OAuth

- ✅ Google OAuth login: existing user masuk tanpa duplicate creation
- ✅ Google OAuth register: new user created dengan proper nonce handling
- ✅ Admin login: hanya email/password, Google OAuth ditolak dengan proper error

- ✅ Edge function check-user-exists: response <500ms, error handling tested
- ✅ Step-up re-auth: aksi sensitif admin memerlukan bukti re-auth <15min
- ✅ Progressive cooldown: 5min → 15min → 60min tested dengan email unlock
- ✅ Anti-enumeration: black-box testing tidak dapat distinguish account existence

Onboarding Flow

- ✅ 5Q verification: jawaban negatif trigger popup konfirmasi keseriusan
- ✅ Batal lanjut: redirect ke Home dengan registrasi gagal message
- ✅ CV wajib form: normalisasi Capitalize Each Word working
- ✅ Onboarding completion: redirect ke /cv-saya sebagai default landing
- ✅ Data tersimpan: onboarding_verifications dengan RLS owner-only verified

CV Management

- ✅ CV preview: hanya field wajib + nama + avatar, empty state handling
- ✅ CV edit: 6 kategori form dengan validation comprehensive
- ✅ Admin note: revisi message display di Edit tab
- ✅ Status tracking: Approve/Revisi reflected dengan kode kandidat generation
- ✅ Normalisasi: semua input menjadi Capitalize Each Word format

Candidate Listing & Search

- ✅ View approved_candidates_v: hanya approved CV tampil di listing
- ✅ Filter functionality: age, education, province dengan auto gender filter untuk lawan jenis
- ✅ Gender auto-filter: user register hanya lihat lawan jenis otomatis
- ✅ Card display: avatar, kode kandidat, pekerjaan, umur, badge status
- ✅ Profile modal: detail CV filtered tanpa PII sensitive
- ✅ Pagination: prev/next working dengan performance acceptable

Taaruf Process

- ✅ Business guards: login + CV approved + koin cukup (sesuai konfigurasi, baseline tunable) + no active taaruf
- ✅ Ajukan taaruf: koin deducted sesuai konfigurasi, status muncul di CV Dikirim
- ✅ CV Masuk: pengajuan dari others dengan aksi terima/tolak
- ✅ Accept/reject: status update real-time dengan notification
- ✅ Taaruf Aktif: kode TAARUF + sequence generated saat accept
- ✅ Active taaruf block: tidak bisa ajukan baru sampai selesai

Payment Integration

- ✅ Midtrans sandbox: integration working dengan semua payment types
- ✅ Webhook signature: SHA512 verification untuk semua callbacks
- ✅ Credit idempotency: order_id + CREDIT, hanya sekali per settlement
- ✅ Debit handling: refund/chargeback dengan order_id + DEBIT idempotency
- ✅ Status matrix: mapping benar per payment_type sesuai dokumentasi
- ✅ Ledger integrity: balance = SUM(ledger_entries), no direct mutations
- ✅ Race protection: concurrent callbacks tidak double-credit/debit

Webhook Security & Resilience

- ✅ Multi-layer throttling: global/IP/order limits active dengan backpressure
- ✅ Rate limiting: baseline tunable dengan runtime adjustment capability
- ✅ Replay protection: ±5min window dengan late replay flagging
- ✅ DLQ processing: failed webhooks queued dan processed via scheduler
- ✅ Forensic logging: correlation-id, request hash, timestamp recorded

Database Performance

- ✅ MV refresh: event-driven enqueue + scheduler execution <1min
- ✅ REFRESH CONCURRENTLY: unique index prerequisites atau fallback documented

- ✅ Size-aware strategy: atomic swap/partition tested tanpa SLA violation
- ✅ Index usage: query performance untuk listing menggunakan proper indexes
- ✅ RLS policies: deny-by-default tested, no data leakage verified

Security Implementation

- ✅ CSP headers: strict policy dengan nonce handling working
- ✅ Input validation: Zod schemas active untuk semua forms, no issue eslint
- ✅ File upload: size/type restrictions dengan security scanning
- ✅ Rate limiting: login/register/actions dengan progressive CAPTCHA
- ✅ Admin security: provider validation, step-up re-auth untuk sensitive ops

Image & Asset Delivery

- ✅ Avatar thumbnails: 20-100KB delivery via Next/Image + CDN
- ✅ Default avatars: <50KB public assets dengan proper caching
- ✅ Private bucket: signed URLs untuk user avatars working
- ✅ Upload validation: <1MB limit dengan type checking enforced
- ✅ Lazy loading: non-critical images dengan performance optimization

Observability & Monitoring

- ✅ X-Request-ID: end-to-end correlation di logs, Sentry, response headers
- ✅ PII-safe logging: audit menunjukkan no sensitive data exposure
- ✅ Error responses: standard format dengan correlation ID untuk support
- ✅ Context propagation: async operations maintain correlation across boundaries
- ✅ Critical operations: payment/webhook/auth/taaruf actions properly tracked

Cost & Operational Management

- ✅ Cost dashboard: menampilkan source, timestamp, confidence indicators
- ✅ Alarm system: 70%/90% thresholds trigger tested response procedures

- ✅ Regional performance: functions di ap-southeast-1, latency <300ms Jakarta
- ✅ Fallback monitoring: alternative data sources working saat API down
- ✅ Scheduled jobs: DLQ/MV refresh/cleanup running dengan SLA compliance

Privacy & Compliance

- ✅ Field exposure: public vs private data matrix enforced
- ✅ Cookie consent: functional dengan categorization working
- ✅ Data retention: policies implemented dengan automated cleanup
- ✅ Export/deletion: user rights workflow functional dan tested

Email & Communication

- ✅ Email deliverability: bounce/block monitoring dengan fallback UX
- ✅ Unlock flow: anti-enumeration protection dengan consistent messaging
- ✅ Transactional emails: welcome, verification, unlock tested dan delivered

Business Logic Validation

- ✅ Kode generation: Ikhwan/Akhwat + sequence untuk approved CV
- ✅ Sequence integrity: TAARUF + sequence untuk active taaruf
- ✅ Status workflows: CV → approved → candidate listing → ajukan → active
- ✅ Balance validation: koin requirements enforced dengan proper error messages
- ✅ Timing constraints: progressive cooldown, step-up re-auth timing working

Performance & SEO

- ✅ Core Web Vitals: LCP/FID/CLS targets met di mobile dan desktop
- ✅ SSR/ISR: proper rendering strategy untuk SEO-critical pages
- ✅ Meta tags: dynamic generation untuk candidate profiles working
- ✅ Sitemap: generated dengan proper candidate URLs included
- ✅ Structured data: JSON-LD untuk rich snippets implemented

Regional & Deployment

- ✅ Functions region: ap-southeast-1 verified via deployment logs
- ✅ CDN performance: global delivery dengan regional optimization
- ✅ Database connections: regional configuration dengan pooling limits
- ✅ Staging/production: parity maintained dengan proper environment separation

Acceptance checklist ini memastikan semua aspek MVP Roomah telah divalidasi dan siap untuk production deployment dengan confidence level tinggi untuk user experience, security, dan business continuity.