

# Mediconnect MVP: Product Requirements Document

---

**Document ID:** PRD-MVP-2025-001  
**Version:** 1.0  
**Date:** September 18, 2025  
**Status:** Final  
**Distribution:** Product Management, Engineering Leadership, Design Team

---

## 0. Executive Summary

---

- **MVP Scope:** Healthcare-as-a-Service platform with GP-orchestrated patient journeys, targeting 500→1,000→10,000 users across 3-month pilot in Nairobi, Kenya
  - **Core Constraint:** 3-month development timeline with phased user scaling, requiring aggressive feature prioritization via MoSCoW framework
  - **Channel Strategy:** Hybrid model combining native Android app for persistent interactions with WhatsApp for authentication (OTP-only) and real-time video consultations
  - **Operating Model:** Closed network HaaS approach where GPs act as clinical front door, curating specialist referrals and pharmacy options rather than open marketplace
  - **Key Differentiators:** AI-powered intake triage, QR-secured digital prescriptions with PDF download disablement, PII-minimized partner portals
  - **Technical Foundation:** PostgreSQL with Row-Level Security, WhatsApp Cloud API integration, mobile-first Android application with web-based partner portals
  - **Localization:** English + Swahili support for patient-facing interfaces, Kenya-specific regulatory compliance and partner network
  - **Security Model:** Least privilege access, comprehensive audit logging, encrypted data at rest/transit, time-boxed support access with PII masking
  - **Primary Risk:** WhatsApp API dependency for critical authentication and consultation flows, requiring robust fallback procedures and SLA monitoring
  - **Success Metrics:** 99.5% platform availability, <500ms P75 API latency, 15-minute GP consult standard with 85%+ completion rate
  - **Out of Scope:** Native video calling, SMS authentication, patient-facing marketplaces, insurance integration, home delivery logistics
  - **Key Assumption:** WhatsApp ubiquity in target market enables seamless user adoption without requiring behavior change for critical interactions
- 

## 1. Consolidated PRD

---

### 1.1 Problem & Goals (MVP)

#### Problem Statement

Healthcare access in Nairobi faces critical barriers: fragmented provider networks, lack of care coordination, unsafe medication dispensing, and limited specialist access. Patients navigate complex systems without clinical guidance, leading to delayed care, inappropriate treatments, and poor health

outcomes. Traditional telemedicine platforms operate as unmanaged marketplaces, failing to ensure care quality or continuity.

### **MVP Goals**

- **Primary:** Establish GP-orchestrated healthcare delivery model serving 10,000 active users within 3-month pilot
- **Clinical:** Achieve 85%+ consultation completion rate with standardized 15-minute GP sessions and structured AI intake
- **Operational:** Build closed partner network of 20+ verified pharmacies and 15+ specialists with <4-hour response SLA
- **Technical:** Deploy secure, scalable platform with 99.5% availability and <500ms API response times
- **User Experience:** Enable seamless care journeys through hybrid app/WhatsApp interface with English/Swahili localization

### **Success Definition**

MVP succeeds when patients can complete end-to-end care journeys (intake → GP consult → prescription/referral → fulfillment) with measurable clinical outcomes, partner satisfaction, and technical reliability supporting planned user scaling.

## **1.2 Personas/Roles**

### **Patient (Primary User)**

- **Demographics:** Urban Nairobi residents, 18-45 years, smartphone users with WhatsApp
- **Needs:** Convenient healthcare access, trusted provider recommendations, transparent pricing, secure health records
- **Behaviors:** Prefers familiar communication channels, values guided experiences over complex choices
- **Pain Points:** Long clinic wait times, difficulty finding specialists, medication authenticity concerns

### **General Practitioner (GP)**

- **Role:** Clinical front door, primary care provider, care coordinator
- **Responsibilities:** Conduct 15-minute consultations, issue prescriptions, manage specialist referrals, maintain clinical notes
- **Tools:** Web portal for patient management, WhatsApp for video consultations, structured intake summaries
- **Success Metrics:** Patient satisfaction scores, consultation completion rates, appropriate referral patterns

### **Specialist**

- **Role:** Expert consultation provider, diagnostics ordering authority
- **Responsibilities:** Provide specialized care via GP referrals, order diagnostic tests, collaborate with referring GP
- **Access Model:** Appointment-based only, no direct patient booking
- **Tools:** Dedicated portal for referral management, diagnostic ordering, results review

### **Pharmacy Admin**

- **Role:** Prescription fulfillment partner
- **Responsibilities:** Verify and dispense medications using QR code system, maintain inventory, report fulfillment status
- **Data Access:** Item-only prescription view with masked patient PII
- **Tools:** Minimal portal for QR scanning and order management

**Diagnostics Admin**

- **Role:** Laboratory services partner
- **Responsibilities:** Process specialist-ordered tests, upload results, maintain quality standards
- **Data Access:** Minimal patient data (name, order ID, tests, masked phone)
- **Tools:** Secure portal for order processing and results upload

**Ops Admin (Internal)**

- **Role:** Platform operations manager
- **Responsibilities:** Manage provider rosters, coordinate complex bookings, handle escalations, onboard partners
- **Access Level:** Full platform access with audit logging
- **Tools:** Comprehensive ops console with calendar views and partner management

**Support (Internal)**

- **Role:** User assistance and technical support
- **Responsibilities:** Resolve user issues, provide platform guidance, escalate complex cases
- **Access Level:** Time-boxed sessions with PII masking by default
- **Tools:** Support portal with masked data views and session logging

**1.3 Scope & Out-of-Scope****In Scope (Must Have)**

- WhatsApp-only authentication via OTP templates
- AI-powered intake questionnaire with GP summary generation
- GP-led video consultations (15min standard, 10min extension for referrals)
- Curated specialist referral workflow with 3-4 provider options
- Digital prescription system with QR/PDF dual format and security controls
- Verified pharmacy network with item-only prescription views
- Specialist-owned diagnostic ordering with minimal PII exposure
- Asynchronous Q&A with 3-6 hour SLA
- Comprehensive notification system (in-app + WhatsApp)
- Row-Level Security with audit logging
- English/Swahili localization for patient interfaces

**In Scope (Should Have)**

- Automated appointment scheduling and rescheduling
- Smart GP rostering with load balancing
- Basic patient health profile (allergies, chronic conditions)
- Enhanced consent management for data sharing
- Advanced ops console with calendar integration
- Pharmacy substitution and dispute handling
- Diagnostic catalogs with cost estimates
- Audio note support in chat threads
- MFA for admin users
- Point-in-time recovery capabilities

**Out of Scope (Won't Have)**

- Native in-app video calling (WhatsApp-only for MVP)
- SMS or email authentication methods
- Patient-facing provider marketplaces
- Direct specialist booking by patients
- Audio/video consultation recording

- Home delivery logistics for medications
- Insurance provider integration and billing
- In-app payment processing
- AI autonomous diagnosis or prescription
- Multi-region deployment beyond Nairobi

## 1.4 Success Metrics

### User Engagement KPIs

- Monthly Active Users: 500 (Month 1) → 1,000 (Month 2) → 10,000 (Month 3)
- Consultation Completion Rate:  $\geq 85\%$  (intake to GP session completion)
- Average Session Duration: 15 minutes  $\pm 2$  minutes for GP consultations
- User Retention:  $\geq 70\%$  30-day retention,  $\geq 50\%$  90-day retention
- Chat Thread Engagement:  $\geq 3$  messages per session average

### Clinical Quality KPIs

- GP Response Time:  $\leq 30$  minutes for consultation requests during business hours
- Specialist Referral Acceptance:  $\geq 80\%$  of GP referrals result in booked appointments
- Prescription Fulfillment:  $\geq 90\%$  of issued prescriptions claimed within 7 days
- Diagnostic Completion:  $\geq 85\%$  of ordered tests completed within SLA timeframes
- Patient Satisfaction:  $\geq 4.2/5.0$  average rating for completed consultations

### Technical Performance KPIs

- Platform Availability:  $\geq 99.5\%$  uptime during pilot period
- API Response Time: P75 latency  $< 500\text{ms}$  for critical endpoints
- Chat Load Time:  $< 1.5$  seconds for last 50 messages
- WhatsApp Integration Reliability:  $\geq 99\%$  successful OTP delivery
- Concurrent User Support: 60+ simultaneous chat sessions without degradation

### Operational Efficiency KPIs

- Partner Network Growth: 20+ verified pharmacies, 15+ specialists by Month 3
- Ops Team Response:  $< 4$  hours for appointment coordination requests
- Support Resolution:  $< 24$  hours for non-critical user issues
- Data Security: Zero PII exposure incidents, 100% audit log coverage
- Cost Per User:  $< \$15$  monthly operational cost per active user

## 1.5 Core User Journeys

### Journey 1: New Patient Onboarding

Narrative: Sarah, a 28-year-old Nairobi resident, downloads the Mediconnect app after a friend's recommendation. She needs to see a doctor for persistent headaches but wants to avoid clinic wait times.

Acceptance Criteria:

- User selects language preference (English/Swahili) on first launch
- Mandatory consent gates for age verification (18+), telemedicine services, and data privacy
- WhatsApp OTP verification with phone number validation
- Profile setup with basic information (name, age, emergency contact)
- Welcome tutorial explaining GP-first care model
- Successful authentication creates secure session with 30-day validity

## **Journey 2: GP Consultation Request**

Narrative: Sarah experiences recurring headaches and decides to consult a GP through Mediconnect. She wants quick access but appreciates the structured intake process.

Acceptance Criteria:

- AI intake questionnaire adapts based on symptom selection
- Structured questions cover onset, severity, duration, associated symptoms
- System generates concise summary for GP review
- Available GP assigned based on roster and specialization
- WhatsApp video call initiated with 15-minute timer
- In-app chat thread remains active for notes and follow-up
- GP can extend session by 10 minutes for referral discussions
- Consultation summary automatically saved to patient record

## **Journey 3: Specialist Referral Process**

Narrative: Sarah's GP determines her headaches require neurological evaluation and initiates a specialist referral during their consultation.

Acceptance Criteria:

- GP discusses referral need and obtains patient consent
- System extends consultation by 10 minutes automatically
- GP presents 3-4 curated neurologist options (in-person/online)
- Patient selects preferred specialist from curated list
- Referral request routed to Ops team for appointment coordination
- Patient receives confirmation with appointment details within 4 hours
- Specialist portal shows referral with GP notes and intake summary
- Appointment reminder sent 24 hours before scheduled time

## **Journey 4: Digital Prescription Fulfillment**

Narrative: Sarah's GP prescribes medication for her headaches. She wants convenient pickup options while ensuring prescription security.

Acceptance Criteria:

- GP issues digital prescription during or after consultation
- Prescription appears in patient app with QR code and PDF download option
- Interactive map displays verified partner pharmacies within 5km radius
- Patient can scan QR at partner pharmacy for instant verification
- Pharmacy portal shows item-only view with masked patient details
- Alternative: PDF download permanently disables QR code
- Fulfillment confirmation updates patient record
- Prescription remains accessible for 90 days in patient history

## **Journey 5: Diagnostic Test Ordering**

Narrative: The neurologist Sarah consulted recommends an MRI scan to rule out underlying conditions. The test must be ordered through the platform's secure process.

Acceptance Criteria:

- Only specialist can initiate diagnostic orders post-consultation
- Order transmitted to verified lab partner with minimal patient data
- Lab portal displays: patient name, order ID, test list, masked phone
- Patient receives notification with lab location and preparation instructions
- Lab uploads results directly to platform upon completion

- Results simultaneously available to ordering specialist and patient
- Specialist reviews results and provides interpretation to patient
- Results archived according to 24-month retention policy

## 1.6 Non-Functional Requirements

### Availability & Reliability

- Platform Uptime: 99.5% availability target during pilot period
- Disaster Recovery: RPO  $\leq 1$  hour, RTO  $\leq 4$  hours for critical services
- WhatsApp Dependency: Documented runbooks for API outages with user communication protocols
- Database Backup: Automated daily backups with point-in-time recovery capability
- Service Monitoring: Real-time alerting for critical path failures

### Performance & Scalability

- User Capacity: Support 10,000 monthly active users by end of pilot
- Concurrent Sessions: Handle 60+ simultaneous chat interactions
- API Latency: P75 response time  $< 500$ ms for authentication, messaging, prescription endpoints
- Chat Performance: Load last 50 messages in  $< 1.5$  seconds
- Database Scaling: Horizontal read replicas for query performance optimization

### Security & Privacy

- Data Encryption: TLS 1.2+ in transit, AES-256 at rest
- Access Control: Row-Level Security policies enforcing least privilege
- PII Minimization: Partner portals expose only necessary data fields
- Audit Logging: Comprehensive event tracking for all data access and modifications
- Session Management: Secure token-based authentication with configurable expiration
- Support Access: Time-boxed sessions with PII masking by default

### Localization & Accessibility

- Language Support: Complete English and Swahili localization for patient interfaces
- Cultural Adaptation: Kenya-specific terminology and healthcare references
- Legal Compliance: Terms of Service and Privacy Policy in English for MVP
- Mobile Optimization: Android-first responsive design with offline capability
- Network Resilience: Graceful degradation for low-bandwidth connections

### Data Management & Retention

- Hot Data Retention: 90 days for chat history and media, 24 months for prescriptions
- Cold Storage: Automated archival to S3-compatible storage (Cloudflare R2)
- Data Recovery: Audited process for restoring archived data on clinical/legal demand
- GDPR Compliance: Right to deletion and data portability mechanisms
- Backup Verification: Monthly restore testing for disaster recovery validation

### Observability & Monitoring

- Application Metrics: Real-time dashboards for user engagement and system health
- Error Tracking: Automated alerting for application exceptions and API failures
- Performance Monitoring: Latency and throughput metrics for critical user journeys
- Business Intelligence: Analytics pipeline for clinical outcomes and operational efficiency
- Compliance Reporting: Automated generation of audit reports for regulatory requirements

## 1.7 Risks & Mitigations

### High-Impact Risks

Risk: WhatsApp API Dependency

- **Impact:** Critical authentication and consultation flows become unavailable
- **Probability:** Medium (external service dependency)
- **Mitigation:** Implement comprehensive monitoring, establish direct Meta support channel, create detailed runbooks for outage scenarios, maintain 48-hour communication buffer for users

Risk: Partner Network Capacity

- **Impact:** Insufficient pharmacy/specialist availability during user scaling
- **Probability:** High (rapid growth scenario)
- **Mitigation:** Aggressive partner onboarding pipeline, capacity monitoring dashboards, overflow protocols with manual ops coordination, geographic expansion of partner network

Risk: Regulatory Compliance Gaps

- **Impact:** Platform shutdown or operational restrictions in Kenya
- **Probability:** Medium (evolving telemedicine regulations)
- **Mitigation:** Engage local healthcare law expertise, establish regulatory monitoring process, build compliance documentation, maintain relationships with Kenya Medical Board

### Medium-Impact Risks

Risk: GP Availability Bottleneck

- **Impact:** Extended wait times leading to user churn
- **Probability:** Medium (limited GP roster)
- **Mitigation:** Smart rostering system, load balancing algorithms, GP recruitment pipeline, asynchronous Q&A as pressure valve

Risk: Data Security Breach

- **Impact:** Patient PII exposure, regulatory penalties, reputation damage
- **Probability:** Low (strong security architecture)
- **Mitigation:** Comprehensive security audits, penetration testing, incident response plan, cyber insurance coverage, staff security training

Risk: Technical Scalability Limits

- **Impact:** Platform performance degradation during user growth
- **Probability:** Medium (aggressive scaling timeline)
- **Mitigation:** Load testing at each scaling milestone, database optimization, CDN implementation, horizontal scaling architecture

### Low-Impact Risks

Risk: User Adoption Resistance

- **Impact:** Slower than projected user growth
- **Probability:** Low (WhatsApp familiarity)
- **Mitigation:** User education campaigns, referral incentives, community health worker partnerships

Risk: Partner Portal Usability Issues

- **Impact:** Operational friction, partner dissatisfaction
  - **Probability:** Medium (complex workflows)
  - **Mitigation:** Extensive partner training, dedicated support channels, iterative UX improvements based on feedback
-

## 2. RBAC Matrix

---



Role	User Mgmt	Con-sults	Mes-sage s	Re-fer-rals	Pre-scrip-tions	Lab Or-ders	Part-ner Data	Ops Con-sole	Audit Logs	Con-strai-nts
<b>Pa-tient</b>	R (own)	CR (own)	CR (own)	R (own)	R (own)	R (own)	-	-	-	Own data only, active ses-sions
<b>GP</b>	R (as-signe d)	CRUD (as-signe d)	CR (as-signe d)	CR (as-signe d)	CR (as-signe d)	R (as-signe d)	R (phar-macy list)	-	-	As-signe d pa-tients only, active roster
<b>Spe-cial-ist</b>	R (re-ferred )	R (re-ferred )	CR (re-ferred )	R (re-ferred )	R (re-ferred )	CRUD (re-ferred )	R (lab list)	-	-	Re-ferred pa-tients only, post-refer-ral
<b>Phar-macy Ad-min</b>	-	-	-	-	R (item s only)	-	RU (own pro-file)	-	-	Item-only view, mask ed PII, own phar-macy
<b>Dia-gnos-tics Ad-min</b>	R (min-imal)	-	-	-	-	RU (as-signe d)	RU (own pro-file)	-	-	Name /Or-orderID/ Tests only, own lab
<b>Ops Ad-min</b>	CRUD	CRUD	R	CRUD	R	R	CRUD	CRUD	R	Full ac-cess,

Role	User Mgmt	Con-sults	Mes-sage s	Re-fer-rals	Pre-scrip-tions	Lab Or-ders	Part-ner Data	Ops Con-sole	Audit Logs	Con-strai-nts
										audit logge d
<b>Sup-port</b>	R (mask ed)	R (mask ed)	R (mask ed)	R (mask ed)	R (mask ed)	R (mask ed)	-	R	-	Time-boxed , PII mask ed, ses-sion logge d

**Legend:** C=Create, R=Read, U=Update, D=Delete, A=Admin

**Key Constraints:**

- All access governed by PostgreSQL Row-Level Security policies
  - Support role requires explicit session initiation with time limits
  - Partner roles limited to own organization's data scope
  - Audit logging mandatory for all admin-level operations
  - PII masking enforced at database query level for restricted roles
-

### **3. Permissioned Page Inventory**

---

#### **Patient Application (Android)**

Screen ID	Screen Name	Roles	Primary Actions	States	Data Access
PAT_001	Language Selection	Patient	Select language, continue	initial, selected	None
PAT_002	Onboarding Consent	Patient	Accept terms, verify age	pending, accepted, rejected	User consent status
PAT_003	Phone Verification	Patient	Enter phone, verify OTP	entering, verifying, verified, failed	Phone number, OTP status
PAT_004	Profile Setup	Patient	Enter details, save profile	editing, saving, complete	Own profile data
PAT_005	Chat Thread	Patient	Send message, view history	active, loading, offline	Own messages, AI responses
PAT_006	AI Intake Form	Patient	Answer questions, submit	in_progress, validating, complete	Own intake responses
PAT_007	Consultation Waiting	Patient	Wait for GP, cancel request	waiting, matched, cancelled	Consultation status
PAT_008	Active Consultation	Patient	Join WhatsApp call, view notes	active, ended	Consultation metadata
PAT_009	Prescription View	Patient	View details, download PDF, find pharmacy	active, downloaded, expired	Own prescriptions
PAT_010	Pharmacy Map	Patient	View locations, get directions	loading, loaded	Verified pharmacy locations
PAT_011	Health Records	Patient	View history, download reports	loading, loaded	Own health records
PAT_012		Patient			

Screen ID	Screen Name	Roles	Primary Actions	States	Data Access
	Appointment Details		View info, reschedule, cancel	scheduled, completed, cancelled	Own appointments

## GP Portal (Web)

Screen ID	Screen Name	Roles	Primary Actions	States	Data Access
GP_001	Login	GP	Authenticate, MFA	authenticating, authenticated, failed	Own credentials
GP_002	Dashboard	GP	View queue, accept consults	loading, active	Assigned patient queue
GP_003	Patient Intake Review	GP	Review AI summary, start consult	reviewing, ready	Patient intake data
GP_004	Consultation Interface	GP	Take notes, prescribe, refer	active, documenting	Patient consultation data
GP_005	Prescription Creation	GP	Select medications, set dosage	creating, validating, issued	Prescription details
GP_006	Referral Management	GP	Select specialists, create referral	selecting, creating, sent	Specialist directory, referral data
GP_007	Patient History	GP	View records, add notes	loading, loaded	Assigned patient history
GP_008	Schedule Management	GP	View roster, set availability	loading, editing, saved	Own schedule data

## Specialist Portal (Web)

Screen ID	Screen Name	Roles	Primary Actions	States	Data Access
SP_001	Login	Specialist	Authenticate, MFA	authenticating, authenticated, failed	Own credentials
SP_002	Referral Queue	Specialist	View referrals, accept appointments	loading, active	Referred patients
SP_003	Referral Details	Specialist	Review GP notes, schedule appointment	reviewing, scheduling	Referral data, GP notes
SP_004	Consultation Interface	Specialist	Conduct consult, order diagnostics	active, documenting	Patient consultation data
SP_005	Diagnostic Ordering	Specialist	Select tests, create orders	selecting, ordering, sent	Test catalog, order details
SP_006	Results Review	Specialist	View results, provide interpretation	loading, reviewing, interpreted	Diagnostic results
SP_007	Patient Communication	Specialist	Send updates, schedule follow-up	composing, sent	Patient communication

## Pharmacy Portal (Web)

Screen ID	Screen Name	Roles	Primary Actions	States	Data Access
PH_001	Login	Pharmacy Admin	Authenticate	authenticating, authenticated, failed	Own credentials
PH_002	QR Scanner	Pharmacy Admin	Scan QR, verify prescription	scanning, verified, invalid	Prescription items only
PH_003	Prescription Details	Pharmacy Admin	View items, mark dispensed	viewing, dispensing, completed	Item list, quantities
PH_004	Inventory Management	Pharmacy Admin	Update stock, set availability	editing, saving	Own inventory data
PH_005	Fulfillment History	Pharmacy Admin	View past orders, generate reports	loading, loaded	Own fulfillment history

## Diagnostics Portal (Web)

Screen ID	Screen Name	Roles	Primary Actions	States	Data Access
DX_001	Login	Diagnostics Admin	Authenticate	authenticating, authenticated, failed	Own credentials
DX_002	Order Queue	Diagnostics Admin	View orders, update status	loading, active	Assigned lab orders
DX_003	Order Details	Diagnostics Admin	View tests, schedule patient	viewing, scheduling	Minimal patient data
DX_004	Results Upload	Diagnostics Admin	Upload files, submit results	uploading, validating, submitted	Test results
DX_005	Order History	Diagnostics Admin	View completed orders	loading, loaded	Own order history

## Ops Console (Web)

Screen ID	Screen Name	Roles	Primary Actions	States	Data Access
OPS_001	Login	Ops Admin	Authenticate, MFA	authenticating, authenticated, failed	Own credentials
OPS_002	Dashboard	Ops Admin	Monitor metrics, view alerts	loading, active	Platform metrics
OPS_003	Provider Roster	Ops Admin	Manage GP/ Specialist schedules	editing, saving	Provider schedules
OPS_004	Appointment Coordination	Ops Admin	Book appointments, resolve conflicts	coordinating, confirmed	Appointment data
OPS_005	Partner Management	Ops Admin	Onboard partners, manage profiles	editing, saving	Partner profiles
OPS_006	Escalation Queue	Ops Admin	Handle complex cases, assign resources	triaging, assigned	Escalated cases
OPS_007	Analytics Dashboard	Ops Admin	View reports, export data	loading, loaded	Aggregated analytics



Support Portal (Web)

Screen ID	Screen Name	Roles	Primary Actions	States	Data Access
SUP_001	Login	Support	Authenticate, start session	authenticating, authenticated, failed	Own credentials
SUP_002	Session Dashboard	Support	View active sessions, time remaining	active, expiring	Session metadata
SUP_003	User Lookup	Support	Search users, view masked data	searching, loaded	Masked user data
SUP_004	Issue Resolution	Support	Document issues, escalate cases	documenting, escalated	Issue details
SUP_005	Knowledge Base	Support	Search articles, view procedures	searching, loaded	Support documentation

4. Task-Centered Information Architecture

Core Task Flows

Task Flow 1: Complete First Consultation

Patient Journey: Download App → Verify Phone → Complete Intake → Consult GP → Receive Prescription

Task Flow 2: Fulfill Prescription

Patient Journey: View Prescription → Find Pharmacy → Present QR Code → Collect Medication

Task Flow 3: Specialist Referral

Patient Journey: GP Consultation → Referral Discussion → Specialist Selection → Appointment Booking → Specialist Consultation

Task Flow 4: Diagnostic Testing

Patient Journey: Specialist Consultation → Test Ordering → Lab Appointment → Test Completion → Results Review

## Platform Sitemap

```
graph TD
    A[Medicconnect Platform] --> B[Patient App]
    A --> C[GP Portal]
    A --> D[Specialist Portal]
    A --> E[Pharmacy Portal]
    A --> F[Diagnostics Portal]
    A --> G[Ops Console]
    A --> H[Support Portal]

    B --> B1[Onboarding]
    B --> B2[Chat Interface]
    B --> B3[Health Records]
    B --> B4[Prescriptions]
    B --> B5[Appointments]

    B1 --> B1a[Language Selection]
    B1 --> B1b[Consent Gates]
    B1 --> B1c[Phone Verification]
    B1 --> B1d[Profile Setup]

    B2 --> B2a[AI Intake]
    B2 --> B2b[Message Thread]
    B2 --> B2c[Consultation Status]

    B4 --> B4a[Prescription Details]
    B4 --> B4b[Pharmacy Map]
    B4 --> B4c[QR Code Display]

    C --> C1[Patient Queue]
    C --> C2[Consultation Interface]
    C --> C3[Prescription Management]
    C --> C4[Referral Creation]
    C --> C5[Schedule Management]

    D --> D1[Referral Queue]
    D --> D2[Consultation Interface]
    D --> D3[Diagnostic Ordering]
    D --> D4[Results Review]

    E --> E1[QR Scanner]
    E --> E2[Prescription Verification]
    E --> E3[Inventory Management]

    F --> F1[Order Queue]
    F --> F2[Results Upload]
    F --> F3[Patient Scheduling]

    G --> G1[Platform Dashboard]
    G --> G2[Provider Management]
    G --> G3[Appointment Coordination]
    G --> G4[Partner Onboarding]

    H --> H1[User Lookup]
    H --> H2[Issue Resolution]
    H --> H3[Session Management]
```

## Screen Inventory by User Journey

### New User Onboarding (4 screens)

- Language Selection → Consent Gates → Phone Verification → Profile Setup

### Consultation Request (6 screens)

- Chat Interface → AI Intake → Consultation Waiting → Active Consultation → Consultation Summary → Follow-up Actions

### Prescription Fulfillment (3 screens)

- Prescription View → Pharmacy Map → QR Code Display

### Specialist Referral (5 screens)

- Referral Discussion → Specialist Selection → Appointment Confirmation → Specialist Consultation → Results Summary

### Health Record Access (2 screens)

- Health Records Dashboard → Detailed Record View

### Provider Workflows (per role)

- GP: 8 core screens for patient management and consultation
  - Specialist: 7 core screens for referral management and diagnostics
  - Pharmacy: 5 core screens for prescription fulfillment
  - Diagnostics: 5 core screens for order processing
  - Ops: 7 core screens for platform management
  - Support: 5 core screens for user assistance
-

## 5. Verbal Wireframes

---

## # Patient Application Screens

### PAT\_001\_language\_selection:

```

screen_id: "PAT_001"
title: "Welcome to Mediconnect"
components:
  - type: "header"
    props:
      logo: "mediconnect_logo"
      subtitle: "Healthcare at your fingertips"
  - type: "language_selector"
    props:
      options: ["English", "Kiswahili"]
      default: "English"
      required: true
  - type: "primary_button"
    props:
      text: "Continue"
      action: "navigate_to_consent"
      enabled: true
states:
  - initial: "language_not_selected"
  - selected: "language_selected"
actions:
  - select_language: "update_language_preference"
  - continue: "navigate_to_PAT_002"

```

### PAT\_002\_onboarding\_consent:

```

screen_id: "PAT_002"
title: "Terms and Consent"
components:
  - type: "progress_indicator"
    props:
      current_step: 1
      total_steps: 4
  - type: "consent_section"
    props:
      title: "Age Verification"
      description: "I confirm that I am 18 years or older"
      required: true
  - type: "consent_section"
    props:
      title: "Telemedicine Services"
      description: "I consent to receive healthcare services via telemedicine"
      required: true
  - type: "consent_section"
    props:
      title: "Data Privacy"
      description: "I agree to the Privacy Policy and Terms of Service"
      required: true
      link: "view_privacy_policy"
  - type: "primary_button"
    props:
      text: "Accept and Continue"
      action: "validate_consents"
      enabled: false
states:
  - pending: "consents_incomplete"
  - accepted: "all_consents_given"
  - rejected: "consent_declined"
actions:
  - toggle_consent: "update_consent_status"

```

- **continue:** "navigate\_to\_PAT\_003"
- **view\_policy:** "open\_privacy\_policy"

#### **PAT\_003\_phone\_verification:**

**screen\_id:** "PAT\_003"

**title:** "Verify Your Phone"

**components:**

- **type:** "progress\_indicator"  
**props:**
  - current\_step:** 2
  - total\_steps:** 4
- **type:** "phone\_input"  
**props:**
  - country\_code:** "+254"
  - placeholder:** "Enter phone number"
  - validation:** "kenya\_mobile\_format"
  - required:** true
- **type:** "info\_card"  
**props:**
  - icon:** "whatsapp\_icon"
  - text:** "We'll send a verification code via WhatsApp"
- **type:** "primary\_button"  
**props:**
  - text:** "Send Code"
  - action:** "request\_otp"
  - enabled:** false
- **type:** "otp\_input"  
**props:**
  - length:** 6
  - visible:** false
  - auto\_focus:** true
- **type:** "secondary\_button"  
**props:**
  - text:** "Resend Code"
  - action:** "resend\_otp"
  - visible:** false
  - cooldown:** 60

**states:**

- **entering:** "phone\_number\_input"
- **verifying:** "otp\_sent\_waiting"
- **verified:** "phone\_verified"
- **failed:** "verification\_failed"

**actions:**

- **enter\_phone:** "validate\_phone\_format"
- **send\_otp:** "trigger\_whatsapp\_otp"
- **verify\_otp:** "validate\_otp\_code"
- **resend:** "resend\_otp\_with\_cooldown"

#### **PAT\_004\_profile\_setup:**

**screen\_id:** "PAT\_004"

**title:** "Complete Your Profile"

**components:**

- **type:** "progress\_indicator"  
**props:**
  - current\_step:** 3
  - total\_steps:** 4
- **type:** "text\_input"  
**props:**
  - label:** "Full Name"
  - placeholder:** "Enter your full name"
  - required:** true
  - validation:** "min\_length\_2"
- **type:** "date\_picker"

```

    props:
      label: "Date of Birth"
      required: true
      max_date: "18_years_ago"
- type: "dropdown"
    props:
      label: "Gender"
      options: ["Male", "Female", "Other", "Prefer not to say"]
      required: false
- type: "text_input"
    props:
      label: "Emergency Contact"
      placeholder: "Emergency contact phone"
      required: true
      validation: "phone_format"
- type: "primary_button"
    props:
      text: "Complete Setup"
      action: "create_profile"
      enabled: false
states:
- editing: "profile_incomplete"
- saving: "profile_creating"
- complete: "profile_created"
actions:
- update_field: "validate_and_update"
- save_profile: "create_user_profile"
- complete: "navigate_to_PAT_005"

PAT_005_chat_thread:
  screen_id: "PAT_005"
  title: "Health Assistant"
  components:
    - type: "header"
      props:
        title: "Chat with AI Assistant"
        subtitle: "Describe your health concern"
        avatar: "ai_assistant_icon"
    - type: "message_list"
      props:
        messages: "chat_history"
        auto_scroll: true
        load_more: "load_older_messages"
        max_visible: 50
    - type: "message_input"
      props:
        placeholder: "Type your message..."
        max_length: 1000
        send_button: true
        attachment_button: false
    - type: "quick_actions"
      props:
        visible: "when_no_active_consultation"
        actions:
          - text: "Start Health Check"
            action: "begin_ai_intake"
          - text: "View My Records"
            action: "navigate_to_health_records"
    - type: "consultation_status"
      props:
        visible: "when_consultation_active"
        status: "consultation_state"
        estimated_wait: "wait_time_minutes"

```

```

states:
  - active: "chat_ready"
  - loading: "messages_loading"
  - offline: "connection_lost"
  - consultation_pending: "waiting_for_gp"
actions:
  - send_message: "post_message_to_thread"
  - start_intake: "navigate_to_PAT_006"
  - join_consultation: "navigate_to_PAT_008"
  - load_history: "fetch_message_history"

PAT_006_ai_intake_form:
  screen_id: "PAT_006"
  title: "Health Assessment"
  components:
    - type: "progress_indicator"
      props:
        current_step: "dynamic"
        total_steps: "dynamic"
        estimated_time: "3-5 minutes"
    - type: "question_card"
      props:
        question_text: "dynamic_question"
        question_type: "multiple_choice|text|scale|date"
        options: "dynamic_options"
        required: true
        help_text: "optional_guidance"
    - type: "navigation_buttons"
      props:
        back_button: "Previous"
        next_button: "Next"
        skip_button: "Skip (if optional)"
    - type: "summary_card"
      props:
        visible: "on_final_step"
        title: "Assessment Summary"
        content: "generated_summary"
    - type: "primary_button"
      props:
        text: "Request GP Consultation"
        action: "submit_intake"
        visible: "on_final_step"
  states:
    - in_progress: "answering_questions"
    - validating: "checking_responses"
    - complete: "intake_finished"
    - error: "validation_failed"
  actions:
    - answer_question: "save_response_and_advance"
    - go_back: "return_to_previous_question"
    - skip_optional: "advance_without_answer"
    - submit: "finalize_intake_and_request_gp"

PAT_007_consultation_waiting:
  screen_id: "PAT_007"
  title: "Finding Your Doctor"
  components:
    - type: "status_card"
      props:
        icon: "doctor_search_animation"
        title: "Connecting you with a GP"
        subtitle: "Estimated wait time: 5-15 minutes"
    - type: "intake_summary"

```



```

    props:
      title: "Your Health Summary"
      content: "ai_generated_summary"
      editable: false
- type: "progress_animation"
  props:
    type: "pulse_dots"
    message: "Finding available doctor..."
- type: "secondary_button"
  props:
    text: "Cancel Request"
    action: "cancel_consultation"
    confirmation_required: true
- type: "info_section"
  props:
    title: "What happens next?"
    items:
      - "A qualified GP will review your summary"
      - "You'll receive a WhatsApp video call"
      - "Consultation typically lasts 15 minutes"
states:
- waiting: "searching_for_gp"
- matched: "gp_assigned"
- cancelled: "request_cancelled"
- timeout: "no_gp_available"
actions:
- cancel: "cancel_consultation_request"
- gp_matched: "navigate_to_PAT_008"
- timeout_reached: "show_reschedule_options"

PAT_008_active_consultation:
screen_id: "PAT_008"
title: "Consultation with Dr. [Name]"
components:
- type: "doctor_card"
  props:
    name: "gp_name"
    specialization: "General Practice"
    avatar: "gp_avatar"
    rating: "gp_rating"
- type: "consultation_timer"
  props:
    duration: "15:00"
    warning_at: "2:00"
    extension_available: true
- type: "whatsapp_call_button"
  props:
    text: "Join Video Call"
    action: "launch_whatsapp_call"
    phone_number: "gp_whatsapp"
    call_type: "video"
- type: "notes_section"
  props:
    title: "Consultation Notes"
    content: "gp_notes"
    real_time_updates: true
    patient_editable: false
- type: "quick_info"
  props:
    items:
      - "Call will start automatically"
      - "Notes will appear here during consultation"
      - "Prescription will be available after consultation"

```

```

states:
  - active: "consultation_in_progress"
  - extended: "consultation_extended"
  - ended: "consultation_completed"
  - disconnected: "call_interrupted"
actions:
  - join_call: "open_whatsapp_video_call"
  - end_consultation: "complete_consultation"
  - request_extension: "extend_consultation_time"

PAT_009_prescription_view:
  screen_id: "PAT_009"
  title: "Your Prescription"
  components:
    - type: "prescription_header"
      props:
        doctor_name: "prescribing_gp"
        date_issued: "prescription_date"
        prescription_id: "unique_id"
    - type: "medication_list"
      props:
        medications: "prescribed_items"
        show_dosage: true
        show_instructions: true
        show_duration: true
    - type: "qr_code_display"
      props:
        qr_data: "prescription_qr"
        size: "large"
        enabled: "qr_active_status"
        disabled_message: "QR code disabled after PDF download"
    - type: "action_buttons"
      props:
        primary_button:
          text: "Find Pharmacy"
          action: "navigate_to_pharmacy_map"
        secondary_button:
          text: "Download PDF"
          action: "download_prescription_pdf"
          warning: "This will disable the QR code"
    - type: "pharmacy_info"
      props:
        title: "Fulfillment Options"
        items:
          - "Scan QR at verified pharmacies"
          - "Download PDF for any pharmacy"
          - "QR code works only once"
  states:
    - active: "qr_enabled"
    - downloaded: "pdf_downloaded_qr_disabled"
    - expired: "prescription_expired"
    - fulfilled: "prescription_completed"
  actions:
    - find_pharmacy: "navigate_to_PAT_010"
    - download_pdf: "generate_and_download_pdf"
    - view_details: "expand_medication_details"

PAT_010_pharmacy_map:
  screen_id: "PAT_010"
  title: "Nearby Pharmacies"
  components:
    - type: "map_view"
      props:

```

```

        center: "user_location"
        zoom_level: 14
        markers: "verified_pharmacies"
        radius: "5km"
-   type: "pharmacy_list"
    props:
        pharmacies: "nearby_verified_pharmacies"
        sort_by: "distance"
        show_distance: true
        show_rating: true
        show_hours: true
-   type: "pharmacy_card"
    props:
        name: "pharmacy_name"
        address: "pharmacy_address"
        distance: "calculated_distance"
        hours: "operating_hours"
        phone: "pharmacy_phone"
        actions:
            - text: "Get Directions"
              action: "open_maps_navigation"
            - text: "Call Pharmacy"
              action: "initiate_phone_call"
-   type: "filter_options"
    props:
        filters:
            - "Open Now"
            - "24 Hours"
            - "Within 2km"

states:
- loading: "fetching_pharmacy_data"
- loaded: "pharmacies_displayed"
- location_denied: "no_location_access"
- no_pharmacies: "no_nearby_pharmacies"

actions:
- select_pharmacy: "show_pharmacy_details"
- get_directions: "launch_navigation_app"
- call_pharmacy: "initiate_phone_call"
- apply_filter: "update_pharmacy_list"

```

# GP Portal Screens

```

GP_001_login:
    screen_id: "GP_001"
    title: "GP Portal Login"
    components:
        - type: "login_form"
          props:
            email_field:
                label: "Email Address"
                required: true
                validation: "email_format"
            password_field:
                label: "Password"
                required: true
                type: "password"
            remember_me:
                label: "Remember me"
                default: false
            login_button:
                text: "Sign In"
                action: "authenticate_gp"
        - type: "mfa_section"

```

```

    props:
      visible: "after_initial_auth"
      title: "Two-Factor Authentication"
      code_input:
        length: 6
        placeholder: "Enter 6-digit code"
      verify_button:
        text: "Verify"
        action: "verify_mfa_code"
  - type: "forgot_password_link"
    props:
      text: "Forgot your password?"
      action: "initiate_password_reset"
states:
  - authenticating: "credentials_validating"
  - mfa_required: "awaiting_mfa_code"
  - authenticated: "login_successful"
  - failed: "authentication_failed"
actions:
  - login: "validate_credentials"
  - verify_mfa: "validate_mfa_token"
  - forgot_password: "send_reset_email"

GP_002_dashboard:
  screen_id: "GP_002"
  title: "GP Dashboard"
  components:
    - type: "header"
      props:
        welcome_message: "Good morning, Dr. [Name]"
        current_time: "system_time"
        status_indicator: "online|offline|busy"
    - type: "patient_queue"
      props:
        title: "Waiting Patients"
        queue_items: "pending_consultations"
        show_wait_time: true
        show_priority: true
        auto_refresh: 30
    - type: "queue_item"
      props:
        patient_name: "masked_patient_name"
        wait_time: "minutes_waiting"
        priority: "normal|urgent"
        summary: "ai_intake_summary"
        actions:
          - text: "Accept"
            action: "accept_consultation"
          - text: "View Details"
            action: "view_patient_intake"
    - type: "stats_cards"
      props:
        cards:
          - title: "Today's Consultations"
            value: "completed_count"
          - title: "Average Duration"
            value: "avg_consultation_time"
          - title: "Queue Length"
            value: "waiting_patients_count"
    - type: "quick_actions"
      props:
        actions:
          - text: "View Schedule"

```

```

        action: "navigate_to_schedule"
      - text: "Patient History"
        action: "navigate_to_patient_search"
    states:
      - loading: "dashboard_loading"
      - active: "queue_monitoring"
      - consultation_active: "in_consultation"
      - offline: "not_available"
    actions:
      - accept_patient: "start_consultation_flow"
      - view_intake: "navigate_to_GP_003"
      - update_status: "change_availability_status"

GP_003_patient_intake_review:
  screen_id: "GP_003"
  title: "Patient Intake Review"
  components:
    - type: "patient_summary"
      props:
        name: "patient_name"
        age: "patient_age"
        gender: "patient_gender"
        phone: "masked_phone"
    - type: "intake_summary"
      props:
        title: "AI-Generated Summary"
        chief_complaint: "primary_concern"
        symptoms: "symptom_list"
        duration: "symptom_duration"
        severity: "severity_scale"
        additional_notes: "patient_notes"
    - type: "medical_history"
      props:
        title: "Relevant History"
        allergies: "known_allergies"
        medications: "current_medications"
        conditions: "chronic_conditions"
        previous_visits: "platform_history"
    - type: "action_buttons"
      props:
        primary_button:
          text: "Start Consultation"
          action: "begin_consultation"
        secondary_button:
          text: "Request More Info"
          action: "send_followup_questions"
        tertiary_button:
          text: "Refer to Specialist"
          action: "direct_referral"
  states:
    - reviewing: "intake_under_review"
    - ready: "ready_for_consultation"
    - consultation_started: "moved_to_consultation"
  actions:
    - start_consult: "navigate_to_GP_004"
    - request_info: "send_patient_message"
    - direct_refer: "navigate_to_GP_006"

GP_004_consultation_interface:
  screen_id: "GP_004"
  title: "Active Consultation"
  components:
    - type: "consultation_header"

```

```

    props:
      patient_name: "patient_name"
      consultation_timer: "15:00"
      extension_available: true
      whatsapp_call_status: "connected|disconnected"
-   type: "notes_editor"
    props:
      title: "Consultation Notes"
      placeholder: "Document your findings..."
      auto_save: true
      save_interval: 30
      templates: "common_note_templates"
-   type: "quick_actions_panel"
    props:
      actions:
        - text: "Prescribe Medication"
          action: "navigate_to_prescription"
          icon: "prescription_icon"
        - text: "Refer to Specialist"
          action: "navigate_to_referral"
          icon: "referral_icon"
        - text: "Schedule Follow-up"
          action: "schedule_followup"
          icon: "calendar_icon"
        - text: "End Consultation"
          action: "complete_consultation"
          icon: "complete_icon"
-   type: "patient_info_sidebar"
    props:
      intake_summary: "condensed_summary"
      medical_history: "relevant_history"
      previous_notes: "past_consultation_notes"
-   type: "consultation_timer"
    props:
      remaining_time: "countdown_timer"
      warning_threshold: "2:00"
      extension_button: "Request 10min Extension"
states:
-   active: "consultation_in_progress"
-   extended: "consultation_extended"
-   completing: "finalizing_notes"
-   completed: "consultation_ended"
actions:
-   save_notes: "auto_save_consultation_notes"
-   prescribe: "navigate_to_GP_005"
-   refer: "navigate_to_GP_006"
-   extend: "request_consultation_extension"
-   complete: "finalize_consultation"

GP_005_prescription_creation:
  screen_id: "GP_005"
  title: "Create Prescription"
  components:
    - type: "patient_context"
      props:
        name: "patient_name"
        age: "patient_age"
        allergies: "known_allergies"
        current_medications: "active_prescriptions"
    - type: "medication_search"
      props:
        search_field:
          placeholder: "Search medications..."

```

```

        autocomplete: true
        drug_database: "kenya_approved_drugs"
        results_list:
            show_generic_name: true
            show_brand_names: true
            show_strength_options: true
    - type: "prescription_builder"
      props:
        selected_medications: "prescription_items"
        dosage_options: "strength_variants"
        frequency_options: ["Once daily", "Twice daily", "Three times daily", "As need
ed"]
        duration_options: ["3 days", "7 days", "14 days", "30 days", "Custom"]
        special_instructions: "free_text_field"
    - type: "drug_interaction_checker"
      props:
        visible: "when_multiple_drugs"
        interactions: "detected_interactions"
        severity_levels: ["Minor", "Moderate", "Major"]
        warnings: "interaction_warnings"
    - type: "prescription_preview"
      props:
        title: "Prescription Summary"
        items: "formatted_prescription"
        patient_instructions: "generated_instructions"
    - type: "action_buttons"
      props:
        primary_button:
            text: "Issue Prescription"
            action: "create_prescription"
        secondary_button:
            text: "Save as Draft"
            action: "save_draft"
        cancel_button:
            text: "Cancel"
            action: "return_to_consultation"
    states:
        - creating: "building_prescription"
        - validating: "checking_interactions"
        - ready: "prescription_complete"
        - issued: "prescription_created"
    actions:
        - search_drug: "query_medication_database"
        - add_medication: "add_to_prescription"
        - check_interactions: "validate_drug_combinations"
        - issue: "finalize_and_issue_prescription"

# Specialist Portal Screens

SP_001_login:
    screen_id: "SP_001"
    title: "Specialist Portal Login"
    components:
        - type: "login_form"
          props:
            email_field:
                label: "Email Address"
                required: true
                validation: "email_format"
            password_field:
                label: "Password"
                required: true
                type: "password"

```

```

    specialty_indicator:
      display: "specialist_type"
      editable: false
    login_button:
      text: "Sign In"
      action: "authenticate_specialist"
  - type: "mfa_section"
    props:
      visible: "after_initial_auth"
      title: "Two-Factor Authentication"
      code_input:
        length: 6
        placeholder: "Enter 6-digit code"
      verify_button:
        text: "Verify"
        action: "verify_mfa_code"
  states:
    - authenticating: "credentials_validating"
    - mfa_required: "awaiting_mfa_code"
    - authenticated: "login_successful"
    - failed: "authentication_failed"
  actions:
    - login: "validate_specialist_credentials"
    - verify_mfa: "validate_mfa_token"

SP_002_referral_queue:
  screen_id: "SP_002"
  title: "Referral Queue"
  components:
    - type: "header"
      props:
        welcome_message: "Dr. [Name] - [Specialty]"
        queue_stats:
          pending_referrals: "count"
          scheduled_appointments: "count"
          completed_today: "count"
    - type: "referral_list"
      props:
        referrals: "pending_referrals"
        sort_options: ["Date Received", "Priority", "Referring GP"]
        filter_options: ["Urgent", "Routine", "Follow-up"]
    - type: "referral_item"
      props:
        patient_name: "masked_patient_name"
        referring_gp: "gp_name"
        referral_date: "date_received"
        priority: "urgent|routine"
        chief_complaint: "referral_reason"
        gp_notes: "referral_summary"
        actions:
          - text: "Review Details"
            action: "view_referral_details"
          - text: "Schedule Appointment"
            action: "schedule_appointment"
          - text: "Decline Referral"
            action: "decline_with_reason"
    - type: "calendar_widget"
      props:
        title: "Today's Appointments"
        appointments: "scheduled_consultations"
        show_time_slots: true
        allow_rescheduling: true
  states:

```



```

- loading: "fetching_referrals"
- active: "queue_displayed"
- appointment_scheduling: "booking_in_progress"
actions:
- view_details: "navigate_to_SP_003"
- schedule: "open_appointment_scheduler"
- decline: "decline_referral_with_reason"

SP_003_referral_details:
screen_id: "SP_003"
title: "Referral Details"
components:
- type: "patient_summary"
  props:
    name: "patient_name"
    age: "patient_age"
    gender: "patient_gender"
    contact: "masked_phone"
- type: "referral_information"
  props:
    referring_gp: "gp_name_and_credentials"
    referral_date: "date_received"
    priority_level: "urgent|routine"
    referral_reason: "chief_complaint"
    clinical_question: "specific_question_for_specialist"
- type: "gp_notes"
  props:
    title: "Referring GP's Notes"
    consultation_summary: "gp_consultation_notes"
    examination_findings: "physical_exam_notes"
    working_diagnosis: "provisional_diagnosis"
    specific_concerns: "gp_concerns"
- type: "patient_history"
  props:
    title: "Relevant Medical History"
    current_medications: "active_prescriptions"
    allergies: "known_allergies"
    chronic_conditions: "ongoing_conditions"
    previous_consultations: "platform_history"
- type: "ai_intake_summary"
  props:
    title: "Original Patient Intake"
    symptoms: "patient_reported_symptoms"
    duration: "symptom_timeline"
    severity: "patient Rated Severity"
    additional_context: "patient_notes"
- type: "action_buttons"
  props:
    primary_button:
      text: "Schedule Appointment"
      action: "schedule_consultation"
    secondary_button:
      text: "Request Additional Info"
      action: "request_more_information"
    tertiary_button:
      text: "Decline Referral"
      action: "decline_with_feedback"
states:
- reviewing: "referral_under_review"
- scheduling: "appointment_being_scheduled"
- scheduled: "appointment_confirmed"
- declined: "referral_declined"
actions:

```

- **schedule:** "open\_appointment\_booking"
- **request\_info:** "send\_information\_request"
- **decline:** "decline\_referral\_with\_reason"

#### # Pharmacy Portal Screens

##### PH\_001\_login:

```

screen_id: "PH_001"
title: "Pharmacy Portal"
components:
  - type: "pharmacy_branding"
    props:
      pharmacy_name: "registered_pharmacy_name"
      logo: "pharmacy_logo"
      license_number: "display_license"
  - type: "login_form"
    props:
      username_field:
        label: "Username"
        required: true
      password_field:
        label: "Password"
        required: true
        type: "password"
      pharmacy_code:
        label: "Pharmacy Code"
        required: true
        validation: "mediconnect_pharmacy_code"
      login_button:
        text: "Sign In"
        action: "authenticate_pharmacy_admin"
  - type: "system_status"
    props:
      mediconnect_status: "online|offline"
      last_sync: "timestamp"
      pending_orders: "count"
states:
  - authenticating: "credentials_validating"
  - authenticated: "login_successful"
  - failed: "authentication_failed"
  - system_offline: "mediconnect_unavailable"
actions:
  - login: "validate_pharmacy_credentials"
  - check_status: "verify_system_connectivity"

```

##### PH\_002\_qr\_scanner:

```

screen_id: "PH_002"
title: "Scan Prescription"
components:
  - type: "camera_scanner"
    props:
      camera_view: "rear_camera"
      scan_overlay: "qr_code_target"
      auto_focus: true
      flash_toggle: true
  - type: "scan_instructions"
    props:
      title: "Scan QR Code"
      instructions:
        - "Point camera at QR code on patient's phone"
        - "Ensure good lighting for clear scan"
        - "QR code will be validated automatically"
  - type: "manual_entry_option"

```

```

    props:
      text: "Enter Code Manually"
      action: "show_manual_input"
      input_field:
        placeholder: "Enter prescription code"
        length: 12
        validation: "alphanumeric"
  - type: "scan_result"
    props:
      visible: "after_successful_scan"
      status: "valid|invalid|expired|already_used"
      prescription_id: "unique_identifier"
      next_action: "proceed_to_verification"
states:
  - scanning: "camera_active"
  - processing: "validating_qr_code"
  - valid: "prescription_verified"
  - invalid: "scan_failed"
  - manual_entry: "entering_code_manually"
actions:
  - scan_qr: "process_qr_code"
  - manual_entry: "validate_manual_code"
  - proceed: "navigate_to_PH_003"

PH_003_prescription_details:
  screen_id: "PH_003"
  title: "Prescription Verification"
  components:
    - type: "prescription_header"
      props:
        prescription_id: "unique_id"
        issue_date: "prescription_date"
        prescribing_doctor: "gp_name"
        verification_status: "verified|pending|expired"
    - type: "medication_list"
      props:
        title: "Prescribed Medications"
        items: "medication_items"
        show_generic_name: true
        show_strength: true
        show_quantity: true
        show_instructions: true
        patient_info_hidden: true
    - type: "medication_item"
      props:
        drug_name: "medication_name"
        strength: "dosage_strength"
        quantity: "prescribed_quantity"
        instructions: "dosage_instructions"
        substitution_allowed: "generic_substitution_flag"
        availability_status: "in_stock|out_of_stock|partial"
    - type: "dispensing_actions"
      props:
        checkboxes:
          - label: "Verify patient identity"
            required: true
          - label: "Counsel patient on usage"
            required: true
          - label: "Check for drug allergies"
            required: true
        substitution_notes:
          visible: "if_substitution_made"
          placeholder: "Note any substitutions made"

```

```

- type: "action_buttons"
  props:
    primary_button:
      text: "Dispense Prescription"
      action: "complete_dispensing"
      enabled: "when_all_checks_complete"
    secondary_button:
      text: "Partial Dispensing"
      action: "dispense_partial"
    cancel_button:
      text: "Cancel"
      action: "return_to_scanner"
  states:
    - verifying: "prescription_being_verified"
    - ready: "ready_for_dispensing"
    - dispensing: "processing_dispensing"
    - completed: "prescription_fulfilled"
    - partial: "partially_dispensed"
  actions:
    - verify_checks: "validate_dispensing_requirements"
    - dispense: "complete_prescription_fulfillment"
    - partial_dispense: "record_partial_fulfillment"

# Diagnostics Portal Screens

DX_001_login:
  screen_id: "DX_001"
  title: "Diagnostics Portal"
  components:
    - type: "lab_branding"
      props:
        lab_name: "registered_lab_name"
        logo: "lab_logo"
        accreditation: "lab_certifications"
    - type: "login_form"
      props:
        username_field:
          label: "Username"
          required: true
        password_field:
          label: "Password"
          required: true
          type: "password"
        lab_code:
          label: "Lab Code"
          required: true
          validation: "mediconnect_lab_code"
        login_button:
          text: "Sign In"
          action: "authenticate_lab_admin"
    - type: "system_status"
      props:
        mediconnect_status: "online|offline"
        pending_orders: "count"
        results_to_upload: "count"
  states:
    - authenticating: "credentials_validating"
    - authenticated: "login_successful"
    - failed: "authentication_failed"
  actions:
    - login: "validate_lab_credentials"

DX_002_order_queue:

```

```

screen_id: "DX_002"
title: "Test Orders"
components:
  - type: "header"
    props:
      lab_name: "laboratory_name"
      queue_stats:
        pending_orders: "count"
        in_progress: "count"
        completed_today: "count"
  - type: "order_list"
    props:
      orders: "pending_lab_orders"
      sort_options: ["Date Received", "Priority", "Test Type"]
      filter_options: ["Urgent", "Routine", "Stat"]
  - type: "order_item"
    props:
      order_id: "unique_order_id"
      patient_name: "patient_name_only"
      ordering_doctor: "specialist_name"
      order_date: "date_received"
      priority: "urgent|routine|stat"
      tests_ordered: "test_list"
      patient_phone: "masked_phone_last_4_digits"
      actions:
        - text: "View Details"
          action: "view_order_details"
        - text: "Schedule Patient"
          action: "schedule_appointment"
        - text: "Mark In Progress"
          action: "update_status_in_progress"
  - type: "quick_stats"
    props:
      cards:
        - title: "Today's Orders"
          value: "daily_order_count"
        - title: "Average TAT"
          value: "average_turnaround_time"
        - title: "Pending Results"
          value: "results_awaiting_upload"
states:
  - loading: "fetching_orders"
  - active: "orders_displayed"
  - scheduling: "appointment_booking"
actions:
  - view_details: "navigate_to_DX_003"
  - schedule: "open_patient_scheduler"
  - update_status: "change_order_status"

DX_003_order_details:
screen_id: "DX_003"
title: "Order Details"
components:
  - type: "order_summary"
    props:
      order_id: "unique_identifier"
      order_date: "date_received"
      priority_level: "urgent|routine|stat"
      ordering_specialist: "doctor_name_and_specialty"
  - type: "patient_info_minimal"
    props:
      patient_name: "full_name_only"
      contact_phone: "masked_phone_number"

```

```

    age_gender: "age_and_gender"
    patient_id: "mediconnect_patient_id"
  - type: "tests_ordered"
    props:
      title: "Requested Tests"
      test_list: "ordered_tests"
      show_test_codes: true
      show_sample_requirements: true
      show_preparation_notes: true
  - type: "test_item"
    props:
      test_name: "diagnostic_test_name"
      test_code: "lab_test_code"
      sample_type: "blood|urine|stool|other"
      preparation_required: "patient_prep_instructions"
      estimated_tat: "turnaround_time"
      status: "pending|collected|processing|completed"
  - type: "clinical_notes"
    props:
      title: "Clinical Information"
      ordering_notes: "specialist_notes"
      clinical_question: "diagnostic_question"
      relevant_history: "patient_context"
  - type: "action_buttons"
    props:
      primary_button:
        text: "Schedule Patient"
        action: "schedule_appointment"
      secondary_button:
        text: "Mark Collected"
        action: "update_sample_collected"
      tertiary_button:
        text: "Upload Results"
        action: "navigate_to_results_upload"
        visible: "when_tests_completed"
states:
  - reviewing: "order_under_review"
  - scheduled: "patient_appointment_set"
  - collected: "samples_collected"
  - processing: "tests_in_progress"
  - ready_for_results: "awaiting_results_upload"
actions:
  - schedule: "open_appointment_scheduler"
  - mark_collected: "update_sample_status"
  - upload_results: "navigate_to_DX_004"

DX_004_results_upload:
  screen_id: "DX_004"
  title: "Upload Results"
  components:
    - type: "order_context"
      props:
        order_id: "unique_identifier"
        patient_name: "patient_name"
        tests_completed: "completed_test_list"
        completion_date: "test_completion_date"
    - type: "file_upload_section"
      props:
        title: "Upload Test Results"
        accepted_formats: ["PDF", "JPEG", "PNG", "DICOM"]
        max_file_size: "10MB per file"
        multiple_files: true
        drag_drop_enabled: true

```

```

- type: "uploaded_files_list"
  props:
    files: "uploaded_result_files"
    show_file_name: true
    show_file_size: true
    show_upload_status: true
    allow_file_removal: true
- type: "results_summary"
  props:
    title: "Results Summary"
    text_area:
      placeholder: "Enter key findings and interpretation..."
      required: true
      max_length: 2000
    critical_values:
      checkbox: "Check if critical values present"
      notification: "Specialist will be notified immediately"
- type: "quality_checks"
  props:
    checkboxes:
      - label: "Results reviewed by qualified personnel"
        required: true
      - label: "All ordered tests completed"
        required: true
      - label: "Patient identity verified"
        required: true
- type: "action_buttons"
  props:
    primary_button:
      text: "Submit Results"
      action: "upload_and_submit_results"
      enabled: "when_all_requirements_met"
    secondary_button:
      text: "Save Draft"
      action: "save_results_draft"
    cancel_button:
      text: "Cancel"
      action: "return_to_order_details"
states:
- uploading: "files_being_uploaded"
- validating: "checking_file_formats"
- ready: "ready_for_submission"
- submitted: "results_uploaded_successfully"
- error: "upload_failed"
actions:
- upload_files: "process_file_uploads"
- submit: "finalize_results_submission"
- save_draft: "save_incomplete_results"

```

---

## 6. Data Model Deltas

---

### Core Tables



```

-- Users and Authentication
CREATE TABLE users (
  id UUID PRIMARY KEY DEFAULT gen_random_uuid(),
  phone_number VARCHAR(15) UNIQUE NOT NULL,
  phone_verified_at TIMESTAMP,
  preferred_language VARCHAR(5) DEFAULT 'en',
  full_name VARCHAR(255),
  date_of_birth DATE,
  gender VARCHAR(20),
  emergency_contact VARCHAR(15),
  created_at TIMESTAMP DEFAULT NOW(),
  updated_at TIMESTAMP DEFAULT NOW(),
  deleted_at TIMESTAMP
);

-- Row-Level Security Policy
ALTER TABLE users ENABLE ROW LEVEL SECURITY;
CREATE POLICY users_own_data ON users FOR ALL USING (id = current_user_id());

-- Healthcare Providers
CREATE TABLE providers (
  id UUID PRIMARY KEY DEFAULT gen_random_uuid(),
  user_id UUID REFERENCES users(id),
  provider_type VARCHAR(20) NOT NULL CHECK (provider_type IN ('gp', 'specialist')),
  specialty VARCHAR(100),
  license_number VARCHAR(50) UNIQUE NOT NULL,
  whatsapp_number VARCHAR(15),
  is_active BOOLEAN DEFAULT true,
  max_concurrent_patients INTEGER DEFAULT 3,
  created_at TIMESTAMP DEFAULT NOW(),
  updated_at TIMESTAMP DEFAULT NOW()
);

ALTER TABLE providers ENABLE ROW LEVEL SECURITY;
CREATE POLICY providers_own_data ON providers FOR ALL USING (user_id = current_user_id());

-- Consultations
CREATE TABLE consultations (
  id UUID PRIMARY KEY DEFAULT gen_random_uuid(),
  patient_id UUID REFERENCES users(id) NOT NULL,
  provider_id UUID REFERENCES providers(id) NOT NULL,
  consultation_type VARCHAR(20) DEFAULT 'gp_primary',
  status VARCHAR(20) DEFAULT 'requested' CHECK (status IN ('requested', 'matched', 'active', 'extended', 'completed', 'cancelled')),
  ai_intake_summary JSONB,
  consultation_notes TEXT,
  started_at TIMESTAMP,
  ended_at TIMESTAMP,
  duration_minutes INTEGER,
  whatsapp_call_initiated BOOLEAN DEFAULT false,
  created_at TIMESTAMP DEFAULT NOW(),
  updated_at TIMESTAMP DEFAULT NOW()
);

ALTER TABLE consultations ENABLE ROW LEVEL SECURITY;
CREATE POLICY consultations_patient_access ON consultations FOR ALL USING (patient_id = current_user_id());
CREATE POLICY consultations_provider_access ON consultations FOR ALL USING (provider_id IN (SELECT id FROM providers WHERE user_id = current_user_id()));

-- Messages (Chat Thread)

```

```

CREATE TABLE messages (
  id UUID PRIMARY KEY DEFAULT gen_random_uuid(),
  consultation_id UUID REFERENCES consultations(id) NOT NULL,
  sender_type VARCHAR(20) NOT NULL CHECK (sender_type IN ('patient', 'ai', 'pro-
vider')),
  sender_id UUID REFERENCES users(id),
  message_text TEXT NOT NULL,
  message_type VARCHAR(20) DEFAULT 'text' CHECK (message_type IN ('text', 'image', '
audio', 'system')),
  metadata JSONB,
  created_at TIMESTAMP DEFAULT NOW()
);

ALTER TABLE messages ENABLE ROW LEVEL SECURITY;
CREATE POLICY messages_consultation_access ON messages FOR ALL USING (
  consultation_id IN (
    SELECT id FROM consultations
    WHERE patient_id = current_user_id()
    OR provider_id IN (SELECT id FROM providers WHERE user_id = current_user_id())
  )
);

-- Prescriptions
CREATE TABLE prescriptions (
  id UUID PRIMARY KEY DEFAULT gen_random_uuid(),
  consultation_id UUID REFERENCES consultations(id) NOT NULL,
  prescribing_provider_id UUID REFERENCES providers(id) NOT NULL,
  patient_id UUID REFERENCES users(id) NOT NULL,
  prescription_data JSONB NOT NULL, -- Medications, dosages, instructions
  qr_code_data VARCHAR(255) UNIQUE NOT NULL,
  qr_enabled BOOLEAN DEFAULT true,
  pdf_downloaded_at TIMESTAMP,
  fulfilled_at TIMESTAMP,
  fulfilling_pharmacy_id UUID REFERENCES partners(id),
  expires_at TIMESTAMP NOT NULL,
  created_at TIMESTAMP DEFAULT NOW(),
  updated_at TIMESTAMP DEFAULT NOW()
);

ALTER TABLE prescriptions ENABLE ROW LEVEL SECURITY;
CREATE POLICY prescriptions_patient_access ON prescriptions FOR ALL USING (patient_id
= current_user_id());
CREATE POLICY prescriptions_provider_access ON prescriptions FOR ALL USING (
  prescribing_provider_id IN (SELECT id FROM providers WHERE user_id = cur-
rent_user_id())
);

-- Referrals
CREATE TABLE referrals (
  id UUID PRIMARY KEY DEFAULT gen_random_uuid(),
  consultation_id UUID REFERENCES consultations(id) NOT NULL,
  referring_provider_id UUID REFERENCES providers(id) NOT NULL,
  patient_id UUID REFERENCES users(id) NOT NULL,
  specialist_id UUID REFERENCES providers(id),
  referral_reason TEXT NOT NULL,
  clinical_question TEXT,
  priority VARCHAR(20) DEFAULT 'routine' CHECK (priority IN ('urgent', 'routine')),
  status VARCHAR(20) DEFAULT 'pending' CHECK (status IN ('pending', 'accepted', 'de-
clined', 'completed')),
  specialist_notes TEXT,
  created_at TIMESTAMP DEFAULT NOW(),
  updated_at TIMESTAMP DEFAULT NOW()
);

```

```

ALTER TABLE referrals ENABLE ROW LEVEL SECURITY;
CREATE POLICY referrals_patient_access ON referrals FOR ALL USING (patient_id = current_user_id());
CREATE POLICY referrals_provider_access ON referrals FOR ALL USING (
    referring_provider_id IN (SELECT id FROM providers WHERE user_id = current_user_id())
    OR specialist_id IN (SELECT id FROM providers WHERE user_id = current_user_id())
);

-- Appointments
CREATE TABLE appointments (
    id UUID PRIMARY KEY DEFAULT gen_random_uuid(),
    referral_id UUID REFERENCES referrals(id),
    patient_id UUID REFERENCES users(id) NOT NULL,
    provider_id UUID REFERENCES providers(id) NOT NULL,
    appointment_type VARCHAR(20) DEFAULT 'specialist_consult',
    scheduled_at TIMESTAMP NOT NULL,
    duration_minutes INTEGER DEFAULT 30,
    status VARCHAR(20) DEFAULT 'scheduled' CHECK (status IN ('scheduled', 'confirmed', 'completed', 'cancelled', 'no_show')),
    consultation_notes TEXT,
    created_at TIMESTAMP DEFAULT NOW(),
    updated_at TIMESTAMP DEFAULT NOW()
);

ALTER TABLE appointments ENABLE ROW LEVEL SECURITY;
CREATE POLICY appointments_patient_access ON appointments FOR ALL USING (patient_id = current_user_id());
CREATE POLICY appointments_provider_access ON appointments FOR ALL USING (
    provider_id IN (SELECT id FROM providers WHERE user_id = current_user_id())
);

-- Lab Orders
CREATE TABLE lab_orders (
    id UUID PRIMARY KEY DEFAULT gen_random_uuid(),
    ordering_provider_id UUID REFERENCES providers(id) NOT NULL,
    patient_id UUID REFERENCES users(id) NOT NULL,
    lab_partner_id UUID REFERENCES partners(id) NOT NULL,
    tests_ordered JSONB NOT NULL, -- Array of test codes and names
    clinical_notes TEXT,
    priority VARCHAR(20) DEFAULT 'routine' CHECK (priority IN ('stat', 'urgent', 'routine')),
    status VARCHAR(20) DEFAULT 'ordered' CHECK (status IN ('ordered', 'scheduled', 'collected', 'processing', 'completed')),
    results_data JSONB,
    results_uploaded_at TIMESTAMP,
    created_at TIMESTAMP DEFAULT NOW(),
    updated_at TIMESTAMP DEFAULT NOW()
);

ALTER TABLE lab_orders ENABLE ROW LEVEL SECURITY;
CREATE POLICY lab_orders_patient_access ON lab_orders FOR ALL USING (patient_id = current_user_id());
CREATE POLICY lab_orders_provider_access ON lab_orders FOR ALL USING (
    ordering_provider_id IN (SELECT id FROM providers WHERE user_id = current_user_id())
);

-- Partners (Pharmacies and Labs)
CREATE TABLE partners (
    id UUID PRIMARY KEY DEFAULT gen_random_uuid(),
    partner_type VARCHAR(20) NOT NULL CHECK (partner_type IN ('pharmacy', 'dia-

```

```

gnostics')),
  name VARCHAR(255) NOT NULL,
  license_number VARCHAR(100) UNIQUE NOT NULL,
  contact_phone VARCHAR(15),
  address TEXT,
  location_coordinates POINT,
  operating_hours JSONB,
  is_active BOOLEAN DEFAULT true,
  created_at TIMESTAMP DEFAULT NOW(),
  updated_at TIMESTAMP DEFAULT NOW()
);

-- Partner Users (Pharmacy/Lab Admins)
CREATE TABLE partner_users (
  id UUID PRIMARY KEY DEFAULT gen_random_uuid(),
  user_id UUID REFERENCES users(id) NOT NULL,
  partner_id UUID REFERENCES partners(id) NOT NULL,
  role VARCHAR(50) NOT NULL,
  is_active BOOLEAN DEFAULT true,
  created_at TIMESTAMP DEFAULT NOW()
);

ALTER TABLE partner_users ENABLE ROW LEVEL SECURITY;
CREATE POLICY partner_users_own_access ON partner_users FOR ALL USING (user_id = current_user_id());

-- Audit Log
CREATE TABLE audit_events (
  id UUID PRIMARY KEY DEFAULT gen_random_uuid(),
  user_id UUID REFERENCES users(id),
  event_type VARCHAR(50) NOT NULL,
  resource_type VARCHAR(50) NOT NULL,
  resource_id UUID NOT NULL,
  event_data JSONB,
  ip_address INET,
  user_agent TEXT,
  created_at TIMESTAMP DEFAULT NOW()
);

-- No RLS on audit_events - only accessible by ops/admin roles

-- Notification Queue
CREATE TABLE notifications (
  id UUID PRIMARY KEY DEFAULT gen_random_uuid(),
  user_id UUID REFERENCES users(id) NOT NULL,
  notification_type VARCHAR(50) NOT NULL,
  title VARCHAR(255) NOT NULL,
  message TEXT NOT NULL,
  channel VARCHAR(20) DEFAULT 'in_app' CHECK (channel IN ('in_app', 'whatsapp', 'bot h')),
  status VARCHAR(20) DEFAULT 'pending' CHECK (status IN ('pending', 'sent', 'delivered', 'failed')),
  metadata JSONB,
  scheduled_at TIMESTAMP DEFAULT NOW(),
  sent_at TIMESTAMP,
  created_at TIMESTAMP DEFAULT NOW()
);

ALTER TABLE notifications ENABLE ROW LEVEL SECURITY;
CREATE POLICY notifications_user_access ON notifications FOR ALL USING (user_id = current_user_id());

```

## Indexes for Performance

```
-- Critical query indexes
CREATE INDEX idx_consultations_patient_status ON consultations(patient_id, status);
CREATE INDEX idx_consultations_provider_status ON consultations(provider_id, status);
CREATE INDEX idx_messages_consultation_created ON messages(consultation_id, created_at);
CREATE INDEX idx_prescriptions_qr_enabled ON prescriptions(qr_code_data) WHERE qr_enabled = true;
CREATE INDEX idx_referrals_specialist_status ON referrals(specialist_id, status);
CREATE INDEX idx_appointments_provider_scheduled ON appointments(provider_id, scheduled_at);
CREATE INDEX idx_lab_orders_partner_status ON lab_orders(lab_partner_id, status);
CREATE INDEX idx_partners_location ON partners USING GIST(location_coordinates);
CREATE INDEX idx_notifications_user_status ON notifications(user_id, status);
CREATE INDEX idx_audit_events_user_created ON audit_events(user_id, created_at);
```

## Data Retention and Archival

```
-- Function to archive old data
CREATE OR REPLACE FUNCTION archive_old_data() RETURNS void AS $$
BEGIN
    -- Archive messages older than 90 days
    INSERT INTO archived_messages
    SELECT * FROM messages
    WHERE created_at < NOW() - INTERVAL '90 days';

    DELETE FROM messages
    WHERE created_at < NOW() - INTERVAL '90 days';

    -- Archive prescriptions older than 24 months
    INSERT INTO archived_prescriptions
    SELECT * FROM prescriptions
    WHERE created_at < NOW() - INTERVAL '24 months';

    DELETE FROM prescriptions
    WHERE created_at < NOW() - INTERVAL '24 months';
END;
$$ LANGUAGE plpgsql;

-- Schedule daily archival
SELECT cron.schedule('archive-old-data', '0 2 * * *', 'SELECT archive_old_data();');
```

## 7. API Stubs

---

```

openapi: 3.0.3
info:
  title: Mediconnect MVP API
  description: Healthcare-as-a-Service platform API for GP-orchestrated patient care
  version: 1.0.0
  contact:
    name: Mediconnect API Team
    email: api-support@mediconnect.health

servers:
- url: https://api.mediconnect.health/v1
  description: Production server
- url: https://staging-api.mediconnect.health/v1
  description: Staging server

security:
- BearerAuth: []

paths:
  # Authentication Endpoints
  /auth/request-otp:
    post:
      summary: Request WhatsApp OTP for authentication
      tags: [Authentication]
      security: []
      requestBody:
        required: true
        content:
          application/json:
            schema:
              type: object
              required: [phone_number]
              properties:
                phone_number:
                  type: string
                  pattern: '^\\+254[0-9]{9}$'
                  example: "+254712345678"
                language:
                  type: string
                  enum: [en, sw]
                  default: en
      responses:
        '200':
          description: OTP sent successfully
          content:
            application/json:
              schema:
                type: object
                properties:
                  success:
                    type: boolean
                    example: true
                  message:
                    type: string
                    example: "OTP sent via WhatsApp"
                  expires_in:
                    type: integer
                    example: 300
        '400':
          $ref: '#/components/responses/BadRequest'
        '429':
          $ref: '#/components/responses/RateLimited'

```

```

/auth/verify-otp:
  post:
    summary: Verify WhatsApp OTP and authenticate user
    tags: [Authentication]
    security: []
    requestBody:
      required: true
      content:
        application/json:
          schema:
            type: object
            required: [phone_number, otp_code]
            properties:
              phone_number:
                type: string
                pattern: '^\\+254[0-9]{9}$'
              otp_code:
                type: string
                pattern: '^[0-9]{6}$'
                example: "123456"
    responses:
      '200':
        description: Authentication successful
        content:
          application/json:
            schema:
              type: object
              properties:
                success:
                  type: boolean
                access_token:
                  type: string
                refresh_token:
                  type: string
                expires_in:
                  type: integer
                  example: 3600
                user:
                  $ref: '#/components/schemas/User'
      '401':
        $ref: '#/components/responses/Unauthorized'

# User Management
/users/profile:
  get:
    summary: Get current user profile
    tags: [Users]
    responses:
      '200':
        description: User profile retrieved
        content:
          application/json:
            schema:
              $ref: '#/components/schemas/User'
      '401':
        $ref: '#/components/responses/Unauthorized'

  put:
    summary: Update user profile
    tags: [Users]
    requestBody:
      required: true

```



```

    content:
      application/json:
        schema:
          type: object
          properties:
            full_name:
              type: string
              maxLength: 255
            date_of_birth:
              type: string
              format: date
            gender:
              type: string
              enum: [male, female, other, prefer_not_to_say]
            emergency_contact:
              type: string
              pattern: '^\\+254[0-9]{9}$'
            preferred_language:
              type: string
              enum: [en, sw]
  responses:
    '200':
      description: Profile updated successfully
      content:
        application/json:
          schema:
            $ref: '#/components/schemas/User'
    '400':
      $ref: '#/components/responses/BadRequest'

# Consultations
/consultations:
  get:
    summary: Get user's consultation history
    tags: [Consultations]
    parameters:
      - name: status
        in: query
        schema:
          type: string
          enum: [requested, matched, active, completed, cancelled]
      - name: limit
        in: query
        schema:
          type: integer
          minimum: 1
          maximum: 50
          default: 20
      - name: offset
        in: query
        schema:
          type: integer
          minimum: 0
          default: 0
    responses:
      '200':
        description: Consultation history retrieved
        content:
          application/json:
            schema:
              type: object
              properties:
                consultations:

```

```

        type: array
        items:
          $ref: '#/components/schemas/Consultation'
      pagination:
        $ref: '#/components/schemas/Pagination'

post:
  summary: Request new GP consultation
  tags: [Consultations]
  requestBody:
    required: true
    content:
      application/json:
        schema:
          type: object
          required: [ai_intake_data]
          properties:
            ai_intake_data:
              type: object
              description: Structured intake responses from AI questionnaire
              properties:
                chief_complaint:
                  type: string
                symptoms:
                  type: array
                  items:
                    type: string
                duration:
                  type: string
                severity:
                  type: integer
                  minimum: 1
                  maximum: 10
                additional_notes:
                  type: string
            priority:
              type: string
              enum: [routine, urgent]
              default: routine
  responses:
    '201':
      description: Consultation request created
      content:
        application/json:
          schema:
            $ref: '#/components/schemas/Consultation'
    '400':
      $ref: '#/components/responses/BadRequest'

/consultations/{consultation_id}:
get:
  summary: Get consultation details
  tags: [Consultations]
  parameters:
    - name: consultation_id
      in: path
      required: true
      schema:
        type: string
        format: uuid
  responses:
    '200':
      description: Consultation details retrieved

```

```

        content:
          application/json:
            schema:
              $ref: '#/components/schemas/Consultation'
      '404':
        $ref: '#/components/responses/NotFound'

/consultations/{consultation_id}/messages:
  get:
    summary: Get consultation message thread
    tags: [Consultations]
    parameters:
      - name: consultation_id
        in: path
        required: true
        schema:
          type: string
          format: uuid
      - name: limit
        in: query
        schema:
          type: integer
          minimum: 1
          maximum: 100
          default: 50
      - name: before
        in: query
        schema:
          type: string
          format: date-time
    responses:
      '200':
        description: Message thread retrieved
        content:
          application/json:
            schema:
              type: object
              properties:
                messages:
                  type: array
                  items:
                    $ref: '#/components/schemas/Message'
                has_more:
                  type: boolean

  post:
    summary: Send message in consultation thread
    tags: [Consultations]
    parameters:
      - name: consultation_id
        in: path
        required: true
        schema:
          type: string
          format: uuid
    requestBody:
      required: true
      content:
        application/json:
          schema:
            type: object
            required: [message_text]
            properties:

```

```

        message_text:
          type: string
          maxLength: 1000
        message_type:
          type: string
          enum: [text, image, audio]
          default: text
      responses:
        '201':
          description: Message sent successfully
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/Message'

# Prescriptions
/prescriptions:
  get:
    summary: Get user's prescriptions
    tags: [Prescriptions]
    parameters:
      - name: status
        in: query
        schema:
          type: string
          enum: [active, fulfilled, expired]
      - name: limit
        in: query
        schema:
          type: integer
          minimum: 1
          maximum: 50
          default: 20
    responses:
      '200':
        description: Prescriptions retrieved
        content:
          application/json:
            schema:
              type: object
              properties:
                prescriptions:
                  type: array
                  items:
                    $ref: '#/components/schemas/Prescription'
                pagination:
                  $ref: '#/components/schemas/Pagination'

/prescriptions/{prescription_id}:
  get:
    summary: Get prescription details
    tags: [Prescriptions]
    parameters:
      - name: prescription_id
        in: path
        required: true
        schema:
          type: string
          format: uuid
    responses:
      '200':
        description: Prescription details retrieved
        content:

```

```

        application/json:
          schema:
            $ref: '#/components/schemas/Prescription'

/prescriptions/{prescription_id}/download:
  post:
    summary: Download prescription PDF (disables QR code)
    tags: [Prescriptions]
    parameters:
      - name: prescription_id
        in: path
        required: true
        schema:
          type: string
          format: uuid
    responses:
      '200':
        description: PDF generated and QR code disabled
        content:
          application/pdf:
            schema:
              type: string
              format: binary
      '410':
        description: PDF already downloaded, QR code disabled
        content:
          application/json:
            schema:
              $ref: '#/components/schemas/Error'

# Pharmacy Integration
/pharmacy/verify-qr:
  post:
    summary: Verify prescription QR code (Pharmacy Portal)
    tags: [Pharmacy]
    security:
      - PharmacyAuth: []
    requestBody:
      required: true
      content:
        application/json:
          schema:
            type: object
            required: [qr_code_data]
            properties:
              qr_code_data:
                type: string
                description: Scanned QR code content
    responses:
      '200':
        description: QR code verified, prescription details returned
        content:
          application/json:
            schema:
              type: object
              properties:
                valid:
                  type: boolean
                  example: true
                prescription:
                  type: object
                  properties:
                    id:

```

```

    type: string
    format: uuid
  medications:
    type: array
    items:
      type: object
      properties:
        name:
          type: string
        strength:
          type: string
        quantity:
          type: string
        instructions:
          type: string
      prescribing_doctor:
        type: string
      issue_date:
        type: string
        format: date-time
'400':
  description: Invalid or expired QR code
  content:
    application/json:
      schema:
        type: object
        properties:
          valid:
            type: boolean
            example: false
          reason:
            type: string
            enum: [invalid_code, expired, already_used, pdf_downloaded]

/pharmacy/fulfill:
  post:
    summary: Mark prescription as fulfilled
    tags: [Pharmacy]
    security:
      - PharmacyAuth: []
    requestBody:
      required: true
      content:
        application/json:
          schema:
            type: object
            required: [prescription_id, dispensed_items]
            properties:
              prescription_id:
                type: string
                format: uuid
              dispensed_items:
                type: array
                items:
                  type: object
                  properties:
                    medication_name:
                      type: string
                    quantity_dispensed:
                      type: string
                    substitution_made:
                      type: boolean
                    substitution_notes:

```

```

        type: string
    verification_checks:
        type: object
    properties:
        patient_identity_verified:
            type: boolean
        counseling_provided:
            type: boolean
        allergies_checked:
            type: boolean
    responses:
        '200':
            description: Prescription marked as fulfilled
            content:
                application/json:
                    schema:
                        type: object
                    properties:
                        success:
                            type: boolean
                        fulfillment_id:
                            type: string
                        format: uuid

# Partner Management
/partners/pharmacies:
    get:
        summary: Get nearby verified pharmacies
        tags: [Partners]
        parameters:
            - name: latitude
              in: query
              required: true
              schema:
                  type: number
                  format: float
            - name: longitude
              in: query
              required: true
              schema:
                  type: number
                  format: float
            - name: radius_km
              in: query
              schema:
                  type: number
                  minimum: 1
                  maximum: 50
                  default: 5
        responses:
            '200':
                description: Nearby pharmacies retrieved
                content:
                    application/json:
                        schema:
                            type: object
                        properties:
                            pharmacies:
                                type: array
                                items:
                                    $ref: '#/components/schemas/Partner'

# Notifications

```

```

/notifications:
  get:
    summary: Get user notifications
    tags: [Notifications]
    parameters:
      - name: status
        in: query
        schema:
          type: string
          enum: [unread, read, all]
          default: all
      - name: limit
        in: query
        schema:
          type: integer
          minimum: 1
          maximum: 50
          default: 20
    responses:
      '200':
        description: Notifications retrieved
        content:
          application/json:
            schema:
              type: object
              properties:
                notifications:
                  type: array
                  items:
                    $ref: '#/components/schemas/Notification'
                unread_count:
                  type: integer

/notifications/{notification_id}/read:
  post:
    summary: Mark notification as read
    tags: [Notifications]
    parameters:
      - name: notification_id
        in: path
        required: true
        schema:
          type: string
          format: uuid
    responses:
      '200':
        description: Notification marked as read
        content:
          application/json:
            schema:
              type: object
              properties:
                success:
                  type: boolean

components:
  securitySchemes:
    BearerAuth:
      type: http
      scheme: bearer
      bearerFormat: JWT
    PharmacyAuth:
      type: http

```



```

scheme: bearer
bearerFormat: JWT
description: Pharmacy partner authentication token

schemas:
  User:
    type: object
    properties:
      id:
        type: string
        format: uuid
      phone_number:
        type: string
      full_name:
        type: string
      date_of_birth:
        type: string
        format: date
      gender:
        type: string
      emergency_contact:
        type: string
      preferred_language:
        type: string
        enum: [en, sw]
      created_at:
        type: string
        format: date-time
      updated_at:
        type: string
        format: date-time

  Consultation:
    type: object
    properties:
      id:
        type: string
        format: uuid
      patient_id:
        type: string
        format: uuid
      provider_id:
        type: string
        format: uuid
      provider_name:
        type: string
      consultation_type:
        type: string
      status:
        type: string
        enum: [requested, matched, active, extended, completed, cancelled]
      ai_intake_summary:
        type: object
      consultation_notes:
        type: string
      started_at:
        type: string
        format: date-time
      ended_at:
        type: string
        format: date-time
      duration_minutes:
        type: integer

```

```

    created_at:
      type: string
      format: date-time

Message:
  type: object
  properties:
    id:
      type: string
      format: uuid
    consultation_id:
      type: string
      format: uuid
    sender_type:
      type: string
      enum: [patient, ai, provider]
    sender_name:
      type: string
    message_text:
      type: string
    message_type:
      type: string
      enum: [text, image, audio, system]
    created_at:
      type: string
      format: date-time

Prescription:
  type: object
  properties:
    id:
      type: string
      format: uuid
    consultation_id:
      type: string
      format: uuid
    prescribing_provider_name:
      type: string
    medications:
      type: array
      items:
        type: object
        properties:
          name:
            type: string
          strength:
            type: string
          quantity:
            type: string
          instructions:
            type: string
    qr_code_data:
      type: string
      description: Only present if QR is still enabled
    qr_enabled:
      type: boolean
    pdf_downloaded_at:
      type: string
      format: date-time
      nullable: true
    fulfilled_at:
      type: string
      format: date-time

```

```

    nullable: true
  expires_at:
    type: string
    format: date-time
  created_at:
    type: string
    format: date-time

```

#### Partner:

```

  type: object
  properties:
    id:
      type: string
      format: uuid
    name:
      type: string
    partner_type:
      type: string
      enum: [pharmacy, diagnostics]
    contact_phone:
      type: string
    address:
      type: string
    latitude:
      type: number
      format: float
    longitude:
      type: number
      format: float
    distance_km:
      type: number
      format: float
    operating_hours:
      type: object
    is_active:
      type: boolean

```

#### Notification:

```

  type: object
  properties:
    id:
      type: string
      format: uuid
    notification_type:
      type: string
    title:
      type: string
    message:
      type: string
    read_at:
      type: string
      format: date-time
      nullable: true
    created_at:
      type: string
      format: date-time

```

#### Pagination:

```

  type: object
  properties:
    total:
      type: integer
    limit:

```

```

    type: integer
  offset:
    type: integer
  has_more:
    type: boolean

Error:
  type: object
  properties:
    error:
      type: object
      properties:
        code:
          type: string
        message:
          type: string
        details:
          type: object

responses:
  BadRequest:
    description: Bad request - invalid input parameters
    content:
      application/json:
        schema:
          $ref: '#/components/schemas/Error'
        example:
          error:
            code: "INVALID_INPUT"
            message: "Phone number format is invalid"

  Unauthorized:
    description: Unauthorized - invalid or missing authentication
    content:
      application/json:
        schema:
          $ref: '#/components/schemas/Error'
        example:
          error:
            code: "UNAUTHORIZED"
            message: "Invalid or expired authentication token"

  NotFound:
    description: Resource not found
    content:
      application/json:
        schema:
          $ref: '#/components/schemas/Error'
        example:
          error:
            code: "NOT_FOUND"
            message: "Consultation not found"

  RateLimited:
    description: Rate limit exceeded
    content:
      application/json:
        schema:
          $ref: '#/components/schemas/Error'
        example:
          error:
            code: "RATE_LIMITED"
            message: "Too many OTP requests. Please wait before trying again."

```

---

## 8. ADRs (Architecture Decision Records)

---

### ADR-001: WhatsApp-Only Authentication

**Status:** Accepted

**Context:** Need reliable, cost-effective authentication for Kenyan market

**Decision:** Use WhatsApp Cloud API for OTP delivery instead of SMS

**Rationale:** Higher delivery rates, lower costs, user familiarity in target market

**Consequences:** Dependency on Meta's infrastructure, requires WhatsApp Business API setup

### ADR-002: Hybrid Communication Model

**Status:** Accepted

**Context:** Balance between controlled app experience and user familiarity

**Decision:** Native app for persistent interactions, WhatsApp for real-time calls

**Rationale:** Leverages existing user behavior while maintaining data control

**Consequences:** Complex integration, dual-channel user experience

### ADR-003: PostgreSQL with Row-Level Security

**Status:** Accepted

**Context:** Need robust data isolation for healthcare compliance

**Decision:** PostgreSQL with RLS policies for multi-tenant data security

**Rationale:** Database-level security enforcement, mature healthcare deployments

**Consequences:** Complex query optimization, requires PostgreSQL expertise

### ADR-004: GP-Gated Specialist Access

**Status:** Accepted

**Context:** Ensure clinical appropriateness and quality control

**Decision:** All specialist referrals must go through GP consultation

**Rationale:** Maintains HaaS model, prevents inappropriate specialist utilization

**Consequences:** Potential bottleneck at GP level, requires robust GP roster management

### ADR-005: QR Code Prescription Security

**Status:** Accepted

**Context:** Prevent prescription fraud and duplicate fulfillment

**Decision:** QR codes disabled permanently after PDF download

**Rationale:** Simple security model, prevents double-dispensing

**Consequences:** User education required, no recovery from accidental downloads

### ADR-006: PII Minimization for Partners

**Status:** Accepted

**Context:** Reduce privacy risk and regulatory compliance burden

**Decision:** Partner portals show only essential data for fulfillment

**Rationale:** Principle of least privilege, reduces breach impact

**Consequences:** Limited partner analytics, potential workflow friction

### ADR-007: Mobile-First Android Development

**Status:** Accepted

**Context:** Resource constraints and market penetration in Kenya

**Decision:** Android-only for MVP, web portals for providers/partners

**Rationale:** Dominant platform in target market, faster development

**Consequences:** Excludes iOS users, requires responsive web design for providers

## ADR-008: Synchronous GP Consultations Only

**Status:** Accepted

**Context:** Ensure real-time clinical decision making

**Decision:** No asynchronous-only GP consultations in MVP

**Rationale:** Clinical safety, relationship building, complex case handling

**Consequences:** GP availability constraints, scheduling complexity

## 9. Open Issues & Decisions Needed

### Build-Blocking Issues (Require Resolution Before Development)

#### Issue #1: WhatsApp Business API Setup

- **Question:** Which WhatsApp Business API provider (Meta direct vs. partner)?
- **Impact:** Authentication flow implementation, cost structure
- **Default:** Use Meta's Cloud API directly for better control and pricing
- **Timeline:** Must resolve by Week 1 of development

#### Issue #2: GP Roster Management

- **Question:** Manual vs. automated GP assignment algorithm?
- **Impact:** Consultation wait times, load balancing
- **Default:** Simple round-robin assignment with manual override capability
- **Timeline:** Must resolve by Week 2 of development

#### Issue #3: Prescription QR Code Format

- **Question:** Custom format vs. standard healthcare QR specification?
- **Impact:** Pharmacy integration complexity, security implementation
- **Default:** Custom JWT-based QR with pharmacy-specific validation
- **Timeline:** Must resolve by Week 1 of development

#### Issue #4: Partner Onboarding Process

- **Question:** Self-service vs. manual partner verification and onboarding?
- **Impact:** Partner network growth rate, operational overhead
- **Default:** Manual verification with standardized onboarding checklist
- **Timeline:** Must resolve by Week 3 of development

#### Issue #5: Data Archival Storage Provider

- **Question:** AWS S3 vs. Cloudflare R2 vs. Google Cloud Storage?
- **Impact:** Long-term storage costs, data retrieval performance
- **Default:** Cloudflare R2 for cost optimization and reduced egress fees
- **Timeline:** Must resolve by Week 4 of development

### Medium-Priority Decisions (Can Be Deferred)

#### Issue #6: AI Intake Model Selection

- **Question:** Custom trained model vs. GPT-4 API vs. rule-based system?
- **Impact:** Intake quality, operational costs, data privacy
- **Default:** Rule-based system with GPT-4 API for summary generation

**Issue #7: Push Notification Provider**

- **Question:** Firebase vs. OneSignal vs. custom implementation?
- **Impact:** Notification reliability, cross-platform support
- **Default:** Firebase Cloud Messaging for Android integration

**Issue #8: Monitoring and Observability Stack**

- **Question:** DataDog vs. New Relic vs. open-source (Prometheus/Grafana)?
- **Impact:** Operational visibility, debugging capabilities, costs
- **Default:** Start with basic logging, upgrade to DataDog post-MVP

**Low-Priority Considerations****Issue #9: Internationalization Framework**

- **Question:** React-i18next vs. custom solution for localization?
- **Impact:** Translation management, future language additions
- **Default:** React-i18next for standardized i18n support

**Issue #10: Error Tracking Service**

- **Question:** Sentry vs. Bugsnag vs. custom error handling?
  - **Impact:** Bug detection, user experience monitoring
  - **Default:** Sentry for comprehensive error tracking and performance monitoring
-

## 10. Changelog

---

### Source Document Mapping



PRD Section	Source Documents	Key Extractions
Executive Summary	Requirements Analysis Executive Summary	MVP scope, constraints, timeline, user scaling
Problem & Goals	Core Product Principles, Patient Journey	HaaS model, GP-orchestrated care, service flows
Personas/Roles	User Roles and Responsibilities	Patient, GP, Specialist, Partner roles and constraints
Scope & Out-of-Scope	MoSCoW Prioritization	Must/Should/Could/Won't have feature categorization
Success Metrics	Non-Functional Requirements	Performance targets, availability, scalability metrics
Core User Journeys	Patient Journey & Core Service Flows	Onboarding, consultation, prescription, referral flows
Non-Functional Requirements	NFRs section	Security, performance, reliability, localization requirements
Risks & Mitigations	Conclusion, Technical Architecture	WhatsApp dependency, partner capacity, regulatory risks
RBAC Matrix	User Roles, Security principles	Role-based access patterns, PII minimization
Permissioned Page Inventory	User Roles, Service Flows	Screen-level access control, role-specific interfaces
Task-Centered IA	Patient Journey, Service Flows	User task flows, navigation patterns, screen relationships
Verbal Wireframes	Service Flows, User Experience	Screen components, states, actions, user interactions
Data Model Deltas	Technical Architecture, Data Model	Entity relationships, RLS policies, audit requirements
API Stubs	Technical Architecture, Integrations	REST endpoints, authentication, partner integrations
ADRs	Technical decisions throughout	Key architectural choices and rationale
Open Issues	Implementation gaps identified	Build-blocking decisions and default recommendations

## Template Compliance

- ✓ **Executive Summary:** 12 bullets covering scope, constraints, risks, assumptions
- ✓ **Consolidated PRD:** 7 subsections with MVP focus and concrete metrics
- ✓ **RBAC Matrix:** Role-capability mapping with CRUD permissions and constraints
- ✓ **Permissioned Page Inventory:** Screen-level access control with states and actions
- ✓ **Task-Centered IA:** User flows with Mermaid sitemap and screen inventory
- ✓ **Verbal Wireframes:** YAML format with stable IDs, components, props, states
- ✓ **Data Model Deltas:** SQL with RLS policies and audit event tracking
- ✓ **API Stubs:** OpenAPI v3 with authentication, pagination, error taxonomy
- ✓ **ADRs:** 8 shortform decisions covering pivotal architectural choices
- ✓ **Open Issues:** 10 build-blocking questions with default recommendations
- ✓ **Changelog:** Source derivation table and template compliance verification

## Version History

- **v1.0** (2025-09-18): Initial PRD creation from consolidated requirements analysis
- **Source:** `/home/ubuntu/mediconnect_requirements_analysis.md` (11 source documents)
- **Scope:** 3-month MVP for Nairobi pilot, 500→1,000→10,000 user scaling
- **Focus:** Build-ready specifications with concrete implementation guidance