**CareNext FRD.**

**1. Version History**

**This document maintains a record of all versions and changes made to the Functional Requirements Document (FRD) for the Carenxt – Patient Appointment & E‑Prescription System.**

* **Version: 1.0**
* **Date:**
* **Description: Initial draft of the FRD for the Carenxt platform, covering all core functional modules.**
* **Author: Omkar Naikade (Business Analyst)**
* **Status: Draft**

**2. Introduction**

This Functional Requirements Document (FRD) describes how the Carenxt – Patient Appointment & E‑Prescription System will work. It explains each feature in detail, including how users interact with the system, what inputs are required, what the system will do, and what outputs will be generated.

The document covers all main functions like patient registration, appointment booking, doctor schedules, consultations, prescriptions, and admin controls. Each section includes the trigger, process, validations, and error handling.

**3. Functional Requirements Overview**

This section provides a list of the key functional requirements for the Carenxt platform. Each requirement addresses a specific function that the system must perform to support patient–doctor interactions through a secure and efficient web interface.

**Below is the list of core functional modules covered in this document:**

* **FR-001 – Patient Registration & Profile Management**
* **FR-002 – Doctor Schedule Setup**
* **FR-003 – Appointment Booking**
* **FR-004 – Video/Clinic Consultation**
* **FR-005 – E-Prescription Generation**
* **FR-006 – Medical History & Report Access**
* **FR-007 – Notification & Reminder System**
* **FR-008 – Admin Dashboard for Clinical Oversight**

**FR-001: Patient Registration & Profile Management**

**Description**  
This function allows new users to register on the Carenxt platform using their mobile number or email. After verification via OTP, users can complete their profile with personal and basic health information. It ensures secure onboarding before they can book appointments.

**Trigger**  
User clicks the “Register” or “Sign Up” button on the Carenxt homepage.

**Actors**  
Patient (User)  
System (Web Platform)

**Preconditions**

* User must not already have an account.
* Mobile number or email must be valid and accessible.
* Internet connection must be active.

**Inputs**

* Mobile number or Email
* OTP
* Password
* Full Name, Date of Birth, Gender
* Address (optional)
* Health details (optional)

**Process / Main Flow**

* User visits the registration page and enters contact details.
* System sends OTP to verify contact.
* On successful OTP verification, the user sets a password.
* User fills in profile information such as name, DOB, gender.
* User can optionally add address and health details.
* System stores the profile and gives access to the dashboard.

**Outputs**

* New user account created and verified
* Completed user profile saved in the system
* Access to appointment booking and history modules

**Validation Rules**

* Email format must be valid
* Mobile number must be 10 digits
* Password must be at least 8 characters
* OTP must match and be valid for 5 minutes

**Error Handling**

* Duplicate email/mobile → “Account already exists”
* Invalid OTP → “Incorrect or expired OTP”
* Weak password → “Password does not meet requirements”
* System error → “Something went wrong. Please try again.”

**Dependencies**

* OTP service provider
* Secure database for user profile storage
* Authentication module

**Alternate Flows**

* If OTP is not received, user can click “Resend OTP”
* If session times out, user can restart registration
* If verification fails 3 times, user is temporarily blocked

**FR-002: Doctor Schedule Setup**

**Description**  
This function allows doctors or clinic admins to set up the doctor’s weekly availability. It includes working days, consultation timings, slot duration, and consultation types (in-clinic or video). The schedule ensures the system can offer time slots to patients for booking.

**Trigger**  
Doctor or admin logs in and opens the “Schedule Setup” section from the dashboard.

**Actors**  
Doctor  
Clinic Admin  
System

**Preconditions**

* Doctor must be verified and approved.
* Admin or doctor must be logged in.
* Scheduling module must be accessible.

**Inputs**

* Available days and working hours
* Slot duration (e.g., 15 or 30 minutes)
* Break times (optional)
* Consultation type (in-clinic, video, or both)
* Holidays or blocked dates

**Process / Main Flow**

* Doctor or admin accesses the schedule setup.
* Selects days and available time ranges.
* Defines slot duration and breaks.
* Chooses consultation types and locations.
* Blocks unavailable dates if required.
* System validates and saves the schedule.

**Outputs**

* Doctor’s availability calendar updated
* Time slots generated and ready for booking
* Schedule stored and shown to patients

**Validation Rules**

* Time slots must not overlap
* Slot duration must be at least 10 minutes
* At least one working day must be selected

**Error Handling**

* Missing data → “Please complete schedule setup”
* Overlapping times → “Slot conflict detected”
* Save failure → “Unable to update. Try again later.”

**Dependencies**

* Calendar and scheduling engine
* Doctor profile and authentication system
* Admin access permissions

**Alternate Flows**

* If doctor has multiple clinics, schedules can be set per location
* If admin updates schedule, changes are logged
* If blocked date overlaps with active slots, system prompts for adjustment

**FR-003: Appointment Booking**

**Description**  
This function allows patients to search for doctors based on speciality, location, and consultation type. Patients can view available slots and book an appointment with their preferred doctor. Bookings can be for in-clinic or video consultations.

**Trigger**  
Patient logs in and clicks on the “Book Appointment” option.

**Actors**  
Patient  
Doctor  
System

**Preconditions**

* Patient must be registered and logged in.
* Doctor must have an active schedule.
* At least one slot must be available.

**Inputs**

* Speciality or doctor name
* Location or consultation type
* Preferred date and time
* Reason for visit (optional)

**Process / Main Flow**

* Patient opens the booking page and uses filters to search for doctors.
* System shows a list of doctors and their available slots.
* Patient selects a doctor and time slot.
* System checks availability and confirms the booking.
* Booking confirmation is shown and shared via email/SMS.

**Outputs**

* Confirmed appointment with doctor, date, and time
* Unique appointment ID
* Notification sent to both patient and doctor

**Validation Rules**

* Slot must be available at the time of booking
* Booking must be at least 1 hour in advance
* One active appointment allowed per patient per doctor

**Error Handling**

* Slot already booked → “This slot is no longer available”
* Duplicate booking → “You already have an upcoming appointment”
* System error → “Booking failed. Please try again”

**Dependencies**

* Doctor schedule module
* Appointment database
* Notification system (SMS/Email)

**Alternate Flows**

* If doctor cancels, patient is notified and offered rescheduling
* If patient cancels, slot is released for others
* If patient misses the appointment, system marks it as “Missed”

**FR-004: Video/Clinic Consultation**

**Description**  
This function allows the doctor and patient to complete a scheduled consultation, either in person (clinic) or through a secure video call. During the consultation, the doctor can review the patient’s history, make notes, and proceed to prescription generation.

**Trigger**  
Scheduled appointment time is reached, and both the doctor and patient join the consultation.

**Actors**  
Patient  
Doctor  
System

**Preconditions**

* A confirmed appointment must exist.
* Both users must be logged in.
* For video calls, a supported browser or device is required.

**Inputs**

* Appointment ID
* Patient medical history (auto-fetched)
* Consultation notes and observations by doctor

**Process / Main Flow**

* Doctor and patient join the consultation at the scheduled time.
* Doctor reviews the patient’s profile and previous records.
* Consultation is conducted (in-clinic or via video).
* Doctor adds notes, diagnosis, and any follow-up instructions.
* System marks the consultation as “Completed”.

**Outputs**

* Updated consultation status
* Notes and diagnosis saved to patient history
* Ready for prescription generation

**Validation Rules**

* Only confirmed appointments can be marked as completed
* Consultation notes must be added before proceeding to prescription
* Video calls must be initiated within the scheduled window

**Error Handling**

* Patient no-show → status set to “Missed”
* Doctor no-show → patient notified and offered rescheduling
* Video call issue → system shows “Unable to connect. Try again.”

**Dependencies**

* Appointment booking module
* Patient medical history
* Video consultation service/API (if applicable)

**Alternate Flows**

* If video call fails, system prompts for switch to audio or reschedule
* If doctor ends early, patient is prompted for feedback
* If patient requests clarification later, doctor can revisit notes

**FR-005: E-Prescription Generation**

**Description**  
This function allows doctors to create and issue a digital prescription after a completed consultation. The prescription includes diagnosis, medication, dosage instructions, and is saved securely to the patient’s medical history.

**Trigger**  
Doctor completes a consultation and clicks on “Generate Prescription”.

**Actors**  
Doctor  
Patient  
System

**Preconditions**

* The consultation must be marked as completed.
* Doctor must be authorised and logged in.
* Patient must have an active profile in the system.

**Inputs**

* Diagnosis and symptoms
* List of prescribed medicines
* Dosage, frequency, and duration
* Additional instructions (if any)

**Process / Main Flow**

* Doctor opens the prescription module after consultation.
* Enters diagnosis, medicines, and instructions.
* System validates input and allows preview.
* Doctor confirms and submits the prescription.
* System generates a signed digital prescription in PDF format.
* Prescription is saved to the patient’s records.

**Outputs**

* Prescription issued and stored
* PDF download available for patient
* Prescription linked to appointment and consultation history

**Validation Rules**

* At least one medication or instruction must be added
* Prescription must be linked to a valid completed appointment
* Only verified doctors can generate prescriptions

**Error Handling**

* Missing diagnosis or medication → “Please complete all fields”
* System error during generation → “Unable to generate prescription. Try again”
* Invalid session → “Session expired. Please re-login”

**Dependencies**

* Completed consultation module
* Patient records and storage system
* Document generation engine (PDF)

**Alternate Flows**

* If doctor chooses to save as draft → prescription status is “In Progress”
* If patient requests edit later → doctor can reissue updated version
* If prescription is cancelled → system marks it as “Cancelled” and logs the reason

**FR-006: Medical History & Report Access**

**Description**  
This function allows patients to view and download their past medical records, including consultation summaries and prescriptions. It also enables doctors to review a patient’s history during future visits.

**Trigger**  
Patient or doctor opens the “Medical History” section from the dashboard.

**Actors**  
Patient  
Doctor  
System

**Preconditions**

* Patient must be registered and logged in.
* At least one past consultation or prescription must exist.
* Data must be stored and accessible.

**Inputs**

* Date range filter (optional)
* Search query by doctor or diagnosis (optional)

**Process / Main Flow**

* User navigates to the “Medical History” page.
* System fetches all relevant records for the logged-in user.
* User can filter, sort, or search records.
* User selects a record to view details or download as PDF.

**Outputs**

* List of consultation and prescription records
* Viewable and downloadable PDFs
* Record of access logged in the system

**Validation Rules**

* Only patients can access their own records
* Records must belong to the authenticated user
* Date format must be valid if filters are applied

**Error Handling**

* No records found → “No medical history available for the selected period”
* Download failure → “Unable to generate file. Please try again”
* Unauthorized access → “You are not allowed to view this data”

**Dependencies**

* Patient record database
* Document storage system
* Secure login and role-based access control

**Alternate Flows**

* If record is archived, system marks it read-only
* If download fails, system offers retry or alternate format
* If session expires, user is logged out and redirected to login

**FR-007: Notification & Reminder System**

**Description**  
This function sends timely notifications and reminders to both patients and doctors about upcoming appointments, new prescriptions, and other important events. Messages are sent via email, SMS, or in-app alerts based on user preferences.

**Trigger**  
A system event occurs such as an appointment being booked, a consultation being completed, or a prescription being issued.

**Actors**  
Patient  
Doctor  
System

**Preconditions**

* Valid contact details (email or mobile) must be available.
* The event must be eligible for notification.
* Notification preferences must be enabled.

**Inputs**

* Event type (appointment, prescription, follow-up, etc.)
* Recipient details (email, mobile number)
* Notification content or template

**Process / Main Flow**

* System detects a trigger event.
* Retrieves relevant user details and preferred channel.
* Generates message using predefined template.
* Sends notification via selected channel (email/SMS).
* Logs notification delivery status.

**Outputs**

* Message delivered to user (email, SMS, or both)
* Record of notification stored in system logs
* Optional status update visible on user dashboard

**Validation Rules**

* Contact details must be valid and verified
* Only approved templates may be used
* Duplicate messages for the same event are suppressed

**Error Handling**

* Invalid contact → “Notification failed: invalid destination”
* SMS/email service error → “Temporary delivery failure”
* System timeout → message queued for retry

**Dependencies**

* External SMS and email service providers
* Event engine for detecting triggers
* User settings and preferences module

**Alternate Flows**

* If notification fails after retries, fallback channel is used
* If patient has opted out, system logs silently without sending
* Admin can manually resend failed messages via dashboard

FR-008: Admin Dashboard for Clinical Oversight Description This function provides administrators with a central dashboard to manage doctor profiles, oversee appointments, monitor system activity, and handle user escalations. It offers tools for reviewing data, generating reports, and ensuring smooth operation of the platform.

Trigger Admin logs into the system and accesses the “Admin Dashboard” section.

Actors Admin System

Preconditions

Admin user must be authenticated with proper access rights.

Doctors and patients must already exist in the system.

Modules for appointments and prescriptions must be active.

Inputs

Doctor and patient activity data

Appointment and consultation records

Filter criteria (date, status, speciality, etc.)

Admin actions (approve, suspend, edit, export)

Process / Main Flow

Admin opens the dashboard and views system summary (doctors, patients, appointments).

Admin can search, filter, and view detailed records.

Admin may take actions such as approving doctors, modifying schedules, or resolving flags.

System logs all actions taken by the admin.

Outputs

Updated doctor and patient records

Exportable reports (appointment stats, doctor activity)

Action logs and audit trail

Validation Rules

Only authorised admin users can access sensitive actions

Mandatory comments required for record changes

Date filters must be valid when generating reports

Error Handling

Permission error → “You do not have access to perform this action”

Report generation failure → “Unable to fetch data. Please try again later”

Invalid input → “Please correct highlighted fields”

Dependencies

User management system

Appointment and consultation modules

Reporting and analytics engine

Alternate Flows

If multiple admins edit the same data, system resolves conflicts

If audit flag is triggered, system alerts super-admin

If system metrics cross threshold, admin receives alerts

**UI/UX requirement**

**4.1 Design Principles**

* Follow a **clean, minimal layout** with user-friendly navigation.
* Use a **responsive web design** that works well on desktop, tablet, and mobile devices.
* Prioritise **accessibility** (clear fonts, colour contrast, keyboard navigation).
* Primary action buttons (e.g., "Book Appointment", "Confirm") should be clearly visible and coloured (e.g., **blue**).
* Use **breadcrumb navigation** or a clear page header to indicate screen context.

**4.2 Input & Form Standards**

* **Labels** should always appear above fields.
* **Inline validation** to provide real-time feedback (e.g., "Invalid mobile number").
* Required fields should have an asterisk \* and appropriate error message.
* Use **dropdowns** for options like gender, consultation type, and specialities.
* Default date pickers for DOB and appointment selection.

**4.3 Button and Link Behaviour**

* Use **primary buttons** for main actions (blue background, white text).  
  Example: “Register”, “Book Appointment”, “Submit Prescription”
* **Secondary buttons** (e.g., “Cancel”, “Back”) should be grey.
* All buttons must have a clear label — avoid icons-only interactions.
* "Resend OTP" or similar time-based buttons should be **disabled for 30 seconds** initially.

**4.4 Error Messages & Feedback**

* Errors should be shown **next to the field**, not as generic banners.
* Use **red for error**, **green for success**, and **orange for warnings**.
* System errors (e.g., server timeout) should be shown in a popup or alert box.

**4.5 Accessibility**

* Font: **Calibri Body**, size minimum 14px
* Text must be resizable (Zoom 125% without breaking layout)
* Icons must include tooltips or alt text
* All forms and buttons must be keyboard-accessible (Tab + Enter)

**User roles and Permissions.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Functionality** | **Patient** | **Doctor** | **Admin** |
| **Register/Login** | Yes | Yes | Yes |
| **Edit Profile** | Yes | Yes | Yes |
| **View Appointments** | Yes | Yes | Yes |
| **Book Appointment** | Yes | No | No |
| **Manage Doctor Schedule** | No | Yes | Yes |
| **Conduct Consultation (In-clinic/Video)** | No | Yes | No |
| **Generate E-Prescription** | No | Yes | No |
| **View Medical History** | Yes (own) | Yes (assigned) | Yes |
| **Access Admin Dashboard** | No | No | Yes |
| **Approve/Deactivate Doctor Profiles** | No | No | Yes |
| **Send/Resend Notifications** | No | No | Yes |
| **Generate Reports** | No | No | Yes |
| **View Audit Logs** | No | No | Yes |

**Error Handling.**

|  |  |  |
| --- | --- | --- |
| **Error Scenario** | **User Message** | **Action / Handling Logic** |
| **Invalid login credentials** | "Incorrect email or password" | Allow retry; lock after 5 failed attempts |
| **OTP expired or incorrect** | "OTP is invalid or has expired" | Allow resend after 30 seconds |
| **Duplicate registration (email or mobile exists)** | "Account already exists with this contact" | Redirect to login or reset password |
| **Required field missing** | "Please fill all mandatory fields" | Highlight missing fields in form |
| **Slot already booked during appointment** | "Selected slot is no longer available" | Prompt user to select another time |
| **Consultation timed out** | "Consultation window has expired" | Allow reschedule or cancel |
| **Prescription generation failed** | "Unable to generate prescription. Try again." | Retry option shown; error logged |
| **Invalid file upload (format or size)** | "File must be JPG, PNG or PDF under 5MB" | Prevent submission and highlight field |
| **Unauthorized access to protected data** | "You do not have permission to view this data" | Redirect to dashboard or login |
| **Server or network error** | "Something went wrong. Please try again later." | Show retry button and log for support |

|  |  |
| --- | --- |
| **Category** | **Requirement** |
| **Performance** | System pages must load within 2 seconds on a 4G connection. |
|  | Search results for appointments must return within 3 seconds. |
|  | Support at least 5,000 concurrent users during peak time. |
| **Security** | All data must be transmitted over HTTPS (TLS 1.2 or higher). |
|  | Passwords must be stored using a strong hashing algorithm (e.g., bcrypt). |
|  | Session timeout after 15 minutes of inactivity. |
|  | Role-based access control must be implemented for all user roles. |
|  | Two-factor authentication (2FA) required for doctors and admins. |
| **Availability** | Platform should maintain 99.5% uptime (excluding maintenance). |
|  | Critical issues must be resolved within 2 hours. |
|  | System data must be backed up every 12 hours. |
| **Usability** | Fully responsive design for web, tablet, and mobile browsers. |
|  | Must comply with WCAG 2.1 Level AA accessibility standards. |
|  | Font size must be at least 14px; form labels must be clearly visible. |
| **Scalability** | System must support scaling up to 1,00,000 users. |
|  | Backend must support modular cloud deployment (e.g., AWS, Azure). |

**System integration requirement**

|  |  |  |  |
| --- | --- | --- | --- |
| **Integration Area** | **External System / API** | **Purpose** | **Module Linked** |
| **OTP Service** | SMS Gateway API (e.g., Twilio, Msg91) | To send OTPs for registration and login verification | FR-001: Patient Registration |
| **Email Notification System** | SMTP / Email Service Provider | To send appointment confirmations and reminders | FR-007: Notification System |
| **Video Consultation API** | Jitsi / Zoom / WebRTC | To enable secure video-based consultations | FR-004: Consultation |
| **Document Storage** | Cloud File Storage (e.g., AWS S3, Firebase) | To store prescriptions, reports, and uploaded documents | FR-005 & FR-006 |
| **Calendar Integration** | Google Calendar API (optional future scope) | Sync doctor schedules with personal calendars | FR-002: Doctor Schedule Setup |
| **Authentication** | Firebase Auth / OAuth2 | Secure login and session handling | All modules |
| **Analytics/Logging** | Logging tool or analytics (e.g., Loggly) | To track system errors, usage, and performance | Admin Module, Global |

**Test case matrix.**

|  |  |  |  |
| --- | --- | --- | --- |
| **FR ID** | **Feature** | **Test Case Range** | **Reason** |
| **FR-001** | **Patient Registration & Profile Management** | 15–20 | Multiple fields, validations, OTP, alternate flows |
| **FR-002** | **Doctor Schedule Setup** | 10–15 | Day/time selection, overlap checks, multiple clinic handling |
| **FR-003** | **Appointment Booking** | 12–18 | Search filters, slot locking, double booking checks, edge case handling |
| **FR-004** | **Video/Clinic Consultation** | 8–12 | Start/end checks, no-show logic, notes capture, video session |
| **FR-005** | **E-Prescription Generation** | 10–14 | Input validation, prescription preview, download checks |
| **FR-006** | **Medical History & Report Access** | 8–10 | Access rights, filter logic, PDF view/download |
| **FR-007** | **Notification & Reminder System** | 6–10 | Trigger conditions, delivery methods, failure handling |
| **FR-008** | **Admin Dashboard for Clinical Oversight** | 12–18 | Role actions, report generation, approval flows, log checks |