

Hospital Management System - Testing Checklist

This comprehensive testing checklist ensures all features of the Hospital Management System work correctly before submission. Test each feature systematically and check off items as you complete them.

Table of Contents

1. Admin Portal Testing
2. Doctor Portal Testing
3. Patient Portal Testing
4. REST API Testing
5. Edge Cases & Validations
6. Email Notifications
7. Performance & UI Testing

Testing Prerequisites

Before you begin testing:

- [] Application is running successfully (`python app.py`)
- [] Database is initialized (`hospital.db` exists)
- [] Browser is open at `http://127.0.0.1:5000`
- [] You have multiple browser tabs ready for different user sessions

A. Admin Portal Testing

Login & Authentication

- [] Navigate to `/admin/login`
- [] Login with correct credentials (username: `admin`, password: `admin123`)
- [] Verify successful login redirects to admin dashboard
- [] Try login with incorrect password - should show error message
- [] Try login with non-existent username - should show error message
- [] Verify admin cannot access doctor portal directly
- [] Verify admin cannot access patient portal directly

Dashboard Overview

- [] View admin dashboard at `/admin/dashboard`
- [] Verify “Total Doctors” count displays correctly

- [] Verify “Total Patients” count displays correctly
- [] Verify “Total Appointments” count displays correctly
- [] Verify “Today’s Appointments” count displays correctly
- [] Check all statistics are updating in real-time
- [] Verify dashboard layout is clean and responsive

Charts & Analytics

- [] View appointments chart/graph on dashboard
- [] Check if chart displays appointments by status (Booked/Completed/Cancelled)
- [] View doctor-wise statistics chart
- [] Verify chart shows number of appointments per doctor
- [] Check if charts are interactive (hover effects, legends)
- [] Verify charts render correctly without JavaScript errors
- [] Test chart responsiveness on different screen sizes

Doctor Management

Add New Doctor

- [] Navigate to “Add Doctor” page
- [] Fill in all required fields:
- [] Full Name (e.g., “Dr. John Smith”)
- [] Email (e.g., “john.smith@hospital.com”)
- [] Password (e.g., “doctor123”)
- [] Specialization (e.g., “Cardiology”)
- [] Qualification (e.g., “MBBS, MD”)
- [] Experience (e.g., “10 years”)
- [] Phone Number (e.g., “9876543210”)
- [] Submit the form
- [] Verify success message appears
- [] Verify new doctor appears in doctors list
- [] Try adding doctor with duplicate email - should show error
- [] Try adding doctor with empty required fields - should show validation errors

View Doctors List

- [] Navigate to “View Doctors” page
- [] Verify all doctors are displayed in a table/list
- [] Check each doctor shows: Name, Specialization, Email, Phone, Actions
- [] Verify pagination works (if implemented)
- [] Verify doctor count matches dashboard statistics

Edit Doctor Details

- [] Click “Edit” button for a doctor
- [] Modify doctor’s name
- [] Modify doctor’s specialization
- [] Modify doctor’s phone number
- [] Save changes
- [] Verify success message

- [] Verify changes are reflected in doctors list
- [] Try saving with invalid email format - should show error
- [] Try saving with empty required fields - should show validation errors

Delete/Blacklist Doctor

- [] Click “Delete” or “Blacklist” button for a doctor
- [] Verify confirmation dialog appears
- [] Confirm deletion
- [] Verify doctor is removed from active list (soft delete)
- [] Verify doctor cannot login after deletion
- [] Check if doctor’s past appointments are still visible
- [] Verify dashboard statistics update correctly

Patient Management

Add New Patient

- [] Navigate to “Add Patient” page
- [] Fill in all required fields:
- [] Full Name (e.g., “Jane Doe”)
- [] Email (e.g., “jane.doe@email.com”)
- [] Password (e.g., “patient123”)
- [] Phone Number (e.g., “9876543210”)
- [] Date of Birth (e.g., “1990-05-15”)
- [] Gender (e.g., “Female”)
- [] Blood Group (e.g., “O+”)
- [] Address (optional)
- [] Submit the form
- [] Verify success message appears
- [] Verify new patient appears in patients list
- [] Try adding patient with duplicate email - should show error

View Patients List

- [] Navigate to “View Patients” page
- [] Verify all patients are displayed
- [] Check each patient shows: Name, Email, Phone, Age, Actions
- [] Verify pagination works (if implemented)
- [] Verify patient count matches dashboard statistics

Edit Patient Details

- [] Click “Edit” button for a patient
- [] Modify patient’s phone number
- [] Modify patient’s address
- [] Modify patient’s blood group
- [] Save changes
- [] Verify success message
- [] Verify changes are reflected in patients list

Delete/Blacklist Patient

- [] Click “Delete” or “Blacklist” button for a patient
- [] Verify confirmation dialog appears
- [] Confirm deletion
- [] Verify patient is removed from active list
- [] Verify patient cannot login after deletion
- [] Check if patient’s past appointments are still visible

Appointment Management

View All Appointments

- [] Navigate to “View Appointments” page
- [] Verify all appointments are displayed
- [] Check each appointment shows:
- [] Patient Name
- [] Doctor Name
- [] Date & Time
- [] Status (Booked/Completed/Cancelled)
- [] Department/Specialization
- [] Verify appointments are sorted by date (latest first or upcoming first)

Filter Appointments

- [] Filter appointments by “Upcoming” - should show only future appointments
- [] Filter appointments by “Past” - should show only completed/past appointments
- [] Filter appointments by “All” - should show all appointments
- [] Filter appointments by “Booked” status
- [] Filter appointments by “Completed” status
- [] Filter appointments by “Cancelled” status
- [] Filter appointments by specific date range (if implemented)
- [] Filter appointments by specific doctor
- [] Filter appointments by specific patient

Search Functionality

Search Doctors

- [] Use search bar to search doctor by full name
- [] Search doctor by partial name (e.g., “John” should find “Dr. John Smith”)
- [] Search doctor by specialization (e.g., “Cardiology”)
- [] Search doctor by email
- [] Verify case-insensitive search works
- [] Verify “No results found” message for invalid searches
- [] Clear search and verify all doctors reappear

Search Patients

- [] Use search bar to search patient by full name
- [] Search patient by partial name
- [] Search patient by phone number
- [] Search patient by email

- [] Search patient by patient ID (if implemented)
- [] Verify case-insensitive search works
- [] Verify “No results found” message for invalid searches
- [] Clear search and verify all patients reappear

Logout

- [] Click “Logout” button in admin portal
 - [] Verify logout confirmation (if implemented)
 - [] Verify redirect to login page or homepage
 - [] Try accessing `/admin/dashboard` after logout - should redirect to login
 - [] Verify session is completely cleared
-

B. Doctor Portal Testing

Login & Authentication

- [] Navigate to `/doctor/login`
- [] Login with doctor credentials (e.g., `dr.sharma` / `doctor123`)
- [] Verify successful login redirects to doctor dashboard
- [] Try login with incorrect password - should show error
- [] Try login with deleted/blacklisted doctor account - should deny access
- [] Verify doctor cannot access admin portal
- [] Verify doctor cannot access patient portal

Dashboard Overview

- [] View doctor dashboard at `/doctor/dashboard`
- [] Verify “Today’s Appointments” section displays appointments for current day
- [] Verify “This Week’s Appointments” section displays appointments for next 7 days
- [] Verify “Total Appointments” count shows all assigned appointments
- [] Check appointment cards show:
 - [] Patient Name
 - [] Date & Time
 - [] Status
 - [] Patient Phone Number
- [] Verify dashboard shows only appointments for the logged-in doctor

Appointment Management

View All Upcoming Appointments

- [] Navigate to “My Appointments” or “All Appointments” page
- [] Verify all upcoming appointments are listed
- [] Verify appointments are sorted by date and time
- [] Check each appointment shows complete details
- [] Filter appointments by “Today”
- [] Filter appointments by “This Week”
- [] Filter appointments by “All Upcoming”

View Past Appointments

- [] Navigate to “Past Appointments” or filter by “Completed”
- [] Verify only completed appointments are shown
- [] Verify completed appointments show treatment details
- [] Verify cancelled appointments are also visible (if implemented)

Mark Appointment as Completed

- [] Find an upcoming appointment with status “Booked”
- [] Click “Complete” or “Mark as Completed” button
- [] Verify confirmation dialog appears
- [] Confirm completion
- [] Verify appointment status changes to “Completed”
- [] Verify appointment moves to past/completed section
- [] Verify dashboard statistics update correctly

Add Treatment Details

- [] Click on a completed appointment
- [] Navigate to “Add Treatment” or “Treatment Details” form
- [] Fill in treatment details:
- [] Diagnosis (e.g., “Hypertension”)
- [] Prescription (e.g., “Amlodipine 5mg - 1 tablet daily”)
- [] Notes (e.g., “Follow-up in 2 weeks”)
- [] Treatment Date (auto-filled)
- [] Save treatment details
- [] Verify success message
- [] Verify treatment details are saved and visible
- [] Try editing treatment details - should allow updates
- [] Verify patient can view these treatment details in their history

Patient Medical History

- [] Click on a patient’s name in appointment list
- [] Navigate to “Patient History” or “Medical Records” page
- [] Verify all past appointments with this patient are displayed
- [] Check each past visit shows:
- [] Date of Visit
- [] Diagnosis
- [] Prescription
- [] Doctor’s Notes
- [] Verify history is sorted chronologically (latest first)
- [] Verify doctor can view full treatment history for informed consultation
- [] Verify “No history available” message for new patients

Availability Management

Set Availability

- [] Navigate to “My Availability” or “Set Availability” page
- [] View current availability for next 7 days

- [] Set availability for a specific date:
- [] Select date
- [] Set time slots (e.g., 9:00 AM - 1:00 PM, 3:00 PM - 6:00 PM)
- [] Or mark as “Not Available”
- [] Save availability settings
- [] Verify success message
- [] Verify availability is visible to patients when booking

Update Availability

- [] Navigate to existing availability settings
- [] Modify time slots for an existing date
- [] Change availability from “Available” to “Not Available”
- [] Change availability from “Not Available” to “Available”
- [] Save changes
- [] Verify changes are reflected immediately
- [] Verify patients see updated availability when booking

Profile Management

- [] Navigate to “My Profile” or “Profile Settings”
- [] View personal information:
 - [] Name
 - [] Email
 - [] Specialization
 - [] Qualification
 - [] Experience
 - [] Phone Number
- [] Click “Edit Profile”
- [] Modify phone number
- [] Modify qualification/experience
- [] Try modifying email (should be allowed or restricted for security)
- [] Save changes
- [] Verify success message
- [] Verify changes are reflected in profile

Logout

- [] Click “Logout” button in doctor portal
- [] Verify redirect to login page
- [] Try accessing `/doctor/dashboard` after logout - should redirect to login
- [] Verify session is cleared

C. Patient Portal Testing

Registration

- [] Navigate to `/patient/register`
- [] Fill in registration form:

- [] Full Name (e.g., "Alice Johnson")
- [] Email (e.g., "alice.johnson@email.com")
- [] Password (minimum 6 characters)
- [] Confirm Password
- [] Phone Number
- [] Date of Birth
- [] Gender
- [] Blood Group (optional)
- [] Address (optional)
- [] Submit registration form
- [] Verify success message
- [] Verify automatic login after registration
- [] Verify redirect to patient dashboard
- [] Try registering with existing email - should show error
- [] Try registering with mismatched passwords - should show error
- [] Try registering with invalid email format - should show validation error

Login & Authentication

- [] Navigate to /patient/login
- [] Login with patient credentials
- [] Verify successful login redirects to patient dashboard
- [] Try login with incorrect password - should show error
- [] Try login with non-existent email - should show error
- [] Verify patient cannot access admin portal
- [] Verify patient cannot access doctor portal

Profile Management

View Profile

- [] Navigate to "My Profile" or "Profile Settings"
- [] Verify all personal information is displayed:
- [] Name
- [] Email
- [] Phone Number
- [] Date of Birth
- [] Age (calculated from DOB)
- [] Gender
- [] Blood Group
- [] Address

Edit Profile

- [] Click "Edit Profile" button
- [] Modify phone number
- [] Modify address
- [] Modify blood group
- [] Try modifying email (should be restricted or require verification)
- [] Try modifying date of birth (should be restricted)

- [] Save changes
- [] Verify success message
- [] Verify changes are reflected immediately in profile

Doctor Search

Search by Specialization

- [] Navigate to “Find Doctors” or “Book Appointment” page
- [] View list of all specializations/departments
- [] Click on a specialization (e.g., “Cardiology”)
- [] Verify only doctors with that specialization are shown
- [] Use search/filter to find specific specialization
- [] Verify search is case-insensitive
- [] Try searching for non-existent specialization - should show “No doctors found”

Search by Doctor Name

- [] Use search bar to find doctor by name
- [] Search by partial name
- [] Verify search results show matching doctors
- [] Verify each doctor card shows:
- [] Name
- [] Specialization
- [] Qualification
- [] Experience
- [] “View Availability” or “Book Appointment” button

View Doctor Details

- [] Click on a doctor card or “View Details” button
- [] View complete doctor profile:
- [] Full Name
- [] Specialization
- [] Qualification
- [] Experience
- [] Available time slots for next 7 days
- [] Verify availability is clearly indicated (available/not available)
- [] Verify “Book Appointment” option is visible for available slots

Appointment Booking

View Doctor Availability

- [] Select a doctor
- [] View availability calendar/schedule for next 7 days
- [] Verify available dates are highlighted or marked clearly
- [] Verify unavailable dates are grayed out or marked as “Not Available”
- [] Verify available time slots are shown for each date

Book Appointment

- [] Select a doctor with available slots
- [] Choose an available date

- [] Choose an available time slot
- [] Add appointment reason/description (optional)
- [] Submit booking request
- [] Verify success message: "Appointment booked successfully"
- [] Verify appointment appears in "My Appointments" list
- [] Verify appointment status is "Booked"
- [] Verify email notification is sent (if email configured)
- [] Try booking same slot again - should show "Slot already booked" error

My Appointments

View Upcoming Appointments

- [] Navigate to "My Appointments" page
- [] View all upcoming appointments
- [] Verify each appointment shows:
- [] Doctor Name
- [] Specialization
- [] Date & Time
- [] Status (Booked)
- [] Appointment ID
- [] Options: Reschedule, Cancel
- [] Verify appointments are sorted by date (earliest first)
- [] Filter appointments by "Upcoming"
- [] Verify only future appointments are shown

Reschedule Appointment

- [] Click "Reschedule" button for an upcoming appointment
- [] Select new date from available slots
- [] Select new time slot
- [] Confirm rescheduling
- [] Verify success message
- [] Verify appointment date/time is updated
- [] Verify old slot becomes available again
- [] Verify email notification is sent (if configured)
- [] Try rescheduling to an already booked slot - should show error

Cancel Appointment

- [] Click "Cancel" button for an upcoming appointment
- [] Verify confirmation dialog: "Are you sure you want to cancel?"
- [] Confirm cancellation
- [] Verify success message: "Appointment cancelled"
- [] Verify appointment status changes to "Cancelled"
- [] Verify cancelled appointment moves to history
- [] Verify time slot becomes available for other patients
- [] Verify email notification is sent (if configured)
- [] Try cancelling already cancelled appointment - should show error or hide option

Appointment History

View Past Appointments

- [] Navigate to “Appointment History” or filter by “Past”
- [] View all completed appointments
- [] Verify each appointment shows:
- [] Doctor Name
- [] Date & Time
- [] Status (Completed)
- [] “View Details” button
- [] Verify appointments are sorted chronologically (latest first)

View Treatment Details

- [] Click on a completed appointment
- [] View treatment details:
- [] Date of Visit
- [] Doctor Name
- [] Diagnosis
- [] Prescription/Medicines
- [] Doctor’s Notes
- [] Follow-up instructions (if any)
- [] Verify all treatment information is displayed correctly
- [] Verify patient can download/print treatment details (if implemented)
- [] Verify “No treatment details available” message if doctor hasn’t added details

View Full Medical History

- [] Navigate to “Medical History” page
- [] View complete medical history across all doctors
- [] Verify history includes:
- [] All past appointments
- [] All diagnoses
- [] All prescriptions
- [] All treatments received
- [] Verify history is sorted chronologically
- [] Verify patient can search/filter medical history
- [] Verify patient can export medical history (if implemented)

Logout

- [] Click “Logout” button in patient portal
 - [] Verify redirect to homepage or login page
 - [] Try accessing /patient/dashboard after logout - should redirect to login
 - [] Verify session is cleared
-

D. REST API Testing

Prerequisites

- [] Install Postman, Insomnia, or use `curl` command
- [] Application is running
- [] You have valid authentication credentials

API Authentication

- [] Test API endpoints without authentication - should return 401 Unauthorized
- [] Test API endpoints with invalid token - should return 401 Unauthorized
- [] Test API endpoints with valid token - should return success

GET Endpoints

GET /api/doctors

- [] Send GET request to `http://127.0.0.1:5000/api/doctors`
- [] Verify response status: 200 OK
- [] Verify response format: JSON
- [] Verify response includes list of doctors
- [] Verify each doctor object contains:
 - [] id
 - [] name
 - [] specialization
 - [] email
 - [] qualification
 - [] experience
- [] Test with query parameters: `/api/doctors?specialization=Cardiology`
- [] Verify pagination (if implemented): `/api/doctors?page=1&limit=10`

Example `curl` command:

```
curl -X GET http://127.0.0.1:5000/api/doctors
```

GET /api/patients

- [] Send GET request to `http://127.0.0.1:5000/api/patients`
- [] Verify requires admin authentication
- [] Verify response status: 200 OK
- [] Verify response format: JSON
- [] Verify response includes list of patients
- [] Verify each patient object contains:
 - [] id
 - [] name
 - [] email
 - [] phone
 - [] age
 - [] gender

Example `curl` command:

```
curl -X GET http://127.0.0.1:5000/api/patients \
-H "Authorization: Bearer YOUR_TOKEN"
```

GET /api/appointments

- [] Send GET request to `http://127.0.0.1:5000/api/appointments`
- [] Verify requires authentication
- [] Verify response status: 200 OK
- [] Verify response format: JSON
- [] Verify response includes list of appointments
- [] Verify each appointment object contains:
 - [] id
 - [] patient_id
 - [] doctor_id
 - [] date
 - [] time
 - [] status
- [] Test with filters: `/api/appointments?status=booked`
- [] Test with date filter: `/api/appointments?date=2025-11-15`

GET /api/stats (Admin Only)

- [] Send GET request to `http://127.0.0.1:5000/api/stats`
- [] Verify requires admin authentication
- [] Verify response status: 200 OK
- [] Verify response includes:
 - [] total_doctors
 - [] total_patients
 - [] total_appointments
 - [] appointments_today
 - [] appointments_by_status (breakdown)
- [] Test access with non-admin user - should return 403 Forbidden

Example curl command:

```
curl -X GET http://127.0.0.1:5000/api/stats \
-H "Authorization: Bearer ADMIN_TOKEN"
```

POST Endpoints

POST /api/doctors (Admin Only)

- [] Send POST request to `http://127.0.0.1:5000/api/doctors`
- [] Set Content-Type header: `application/json`
- [] Send request body:

```
{
  "name": "Dr. Test User",
  "email": "test.doctor@hospital.com",
  "password": "testpass123",
  "specialization": "Dermatology",
  "qualification": "MBBS, MD",
  "experience": "5 years",
  "phone": "9876543210"
}
```

- [] Verify response status: 201 Created
- [] Verify response includes created doctor object
- [] Verify doctor is added to database
- [] Try creating duplicate email - should return 400 Bad Request
- [] Try with missing required fields - should return 400 Bad Request

Example curl command:

```
curl -X POST http://127.0.0.1:5000/api/doctors \
-H "Content-Type: application/json" \
-H "Authorization: Bearer ADMIN_TOKEN" \
-d '{"name": "Dr. Test", "email": "test@hospital.com", "password": "test123", "specialization": "Dermatology", "qualification": "MBBS", "experience": "5", "phone": "9876543210"}'
```

POST /api/appointments

- [] Send POST request to `http://127.0.0.1:5000/api/appointments`
- [] Set Content-Type header: `application/json`
- [] Send request body:

```
{
  "patient_id": 1,
  "doctor_id": 2,
  "date": "2025-11-20",
  "time": "10:00",
  "reason": "Regular checkup"
}
```

- [] Verify response status: 201 Created
- [] Verify response includes created appointment object
- [] Verify appointment is added to database
- [] Try booking same slot - should return 400 Bad Request
- [] Try booking in the past - should return 400 Bad Request
- [] Try with invalid doctor_id - should return 404 Not Found

Example curl command:

```
curl -X POST http://127.0.0.1:5000/api/appointments \
-H "Content-Type: application/json" \
-H "Authorization: Bearer PATIENT_TOKEN" \
-d '{"patient_id": 1, "doctor_id": 2, "date": "2025-11-20", "time": "10:00", "reason": "Checkup"}'
```

PUT/PATCH Endpoints (if implemented)

UPDATE /api/appointments/:id

- [] Send PUT/PATCH request to `http://127.0.0.1:5000/api/appointments/1`
- [] Update appointment status to “Completed”
- [] Verify response status: 200 OK
- [] Verify appointment is updated in database
- [] Try updating with invalid status - should return 400 Bad Request

DELETE Endpoints (if implemented)

DELETE /api/appointments/:id

- [] Send DELETE request to `http://127.0.0.1:5000/api/appointments/1`
- [] Verify response status: 200 OK or 204 No Content
- [] Verify appointment is deleted/cancelled in database
- [] Try deleting non-existent appointment - should return 404 Not Found

Error Handling

- [] Test API with malformed JSON - should return 400 Bad Request
 - [] Test API with wrong HTTP method - should return 405 Method Not Allowed
 - [] Test API with non-existent endpoint - should return 404 Not Found
 - [] Verify all error responses include meaningful error messages
 - [] Verify all error responses are in JSON format
-

E. Edge Cases & Validations

Double Booking Prevention

- [] Login as two different patients in two browser tabs
- [] Both try to book the same doctor at the same date/time simultaneously
- [] Verify only one booking succeeds
- [] Verify the other gets “Slot already booked” error
- [] Verify database integrity (no duplicate bookings)

Past Date Validation

- [] Try booking appointment for a date in the past
- [] Verify error message: “Cannot book appointment in the past”
- [] Try rescheduling to a past date
- [] Verify system prevents past date selection
- [] Verify date picker disables past dates

Invalid Login Attempts

- [] Try admin login with patient credentials - should fail
- [] Try doctor login with admin credentials - should fail
- [] Try patient login with doctor credentials - should fail
- [] Try logging in with SQL injection attempts (e.g., `' OR '1'='1`)
- [] Verify system prevents unauthorized access

- [] Verify proper error messages without revealing security info

Unauthorized Access Prevention

- [] Logout from patient portal
- [] Try accessing `/patient/dashboard` directly via URL
- [] Verify redirect to login page
- [] Try accessing `/admin/dashboard` as a patient
- [] Verify “Access Denied” or redirect to patient portal
- [] Try accessing `/doctor/dashboard` as a patient
- [] Verify “Access Denied” or redirect to patient portal
- [] Try accessing other users’ data via URL manipulation (e.g., `/patient/profile/2`)
- [] Verify users can only access their own data

Form Validation

Empty Fields

- [] Submit login form with empty username
- [] Submit login form with empty password
- [] Submit registration form with missing required fields
- [] Submit doctor add form with empty specialization
- [] Verify appropriate error messages appear
- [] Verify form highlights invalid fields

Invalid Data Formats

- [] Enter invalid email format (e.g., “notanemail”)
- [] Enter phone number with letters (e.g., “abcd1234”)
- [] Enter phone number with less than 10 digits
- [] Enter password less than minimum length
- [] Enter future date of birth
- [] Verify validation errors appear
- [] Verify form doesn’t submit with invalid data

Password Mismatch

- [] Register patient with mismatched password and confirm password
- [] Change password with mismatched new password and confirm
- [] Verify error message: “Passwords do not match”
- [] Verify form doesn’t submit

Session Management

- [] Login in one browser tab
- [] Logout in another browser tab
- [] Try performing action in first tab
- [] Verify user is logged out everywhere
- [] Test session timeout (if implemented)
- [] Verify session persistence after browser close (if “Remember Me” is implemented)

Responsive Design Testing

- [] Open application on desktop (1920x1080)

- [] Verify layout looks clean and professional
- [] Open application on tablet size (768x1024)
- [] Verify layout adjusts properly
- [] Open application on mobile size (375x667)
- [] Verify layout is mobile-friendly
- [] Test navigation menu on mobile (hamburger menu if implemented)
- [] Test forms on mobile devices
- [] Test tables on mobile (should scroll or stack)
- [] Verify all buttons are easily clickable on touch screens

Data Integrity

- [] Delete a doctor who has upcoming appointments
- [] Verify appointments are handled correctly (cancelled or reassigned)
- [] Delete a patient who has past appointments
- [] Verify appointment history is preserved
- [] Complete an appointment and add treatment
- [] Delete the doctor later
- [] Verify treatment history remains accessible to patient

Concurrent Operations

- [] Login as admin in one tab
 - [] Login as doctor in another tab
 - [] Admin edits doctor's details
 - [] Doctor tries to update profile simultaneously
 - [] Verify no data conflicts occur
 - [] Verify latest changes are saved correctly
-

F. Email Notifications (If Configured)

Appointment Booking Email

- [] Book a new appointment as patient
- [] Check patient's email inbox
- [] Verify appointment confirmation email received
- [] Verify email contains:
- [] Patient name
- [] Doctor name
- [] Appointment date and time
- [] Department/Specialization
- [] Appointment ID
- [] Hospital contact information
- [] Check doctor's email inbox (if doctor notifications are enabled)
- [] Verify doctor receives new appointment notification

Appointment Cancellation Email

- [] Cancel an appointment as patient
- [] Check patient's email inbox
- [] Verify cancellation confirmation email received
- [] Verify email contains:
- [] Appointment details
- [] Cancellation confirmation
- [] Refund policy (if applicable)
- [] Check doctor's email inbox
- [] Verify doctor receives cancellation notification

Appointment Rescheduling Email

- [] Reschedule an appointment as patient
- [] Check patient's email inbox
- [] Verify rescheduling confirmation email received
- [] Verify email shows old and new date/time
- [] Check doctor's email inbox
- [] Verify doctor receives rescheduling notification

Email Formatting

- [] Verify emails have professional HTML formatting
 - [] Verify emails have proper subject lines
 - [] Verify emails have hospital logo (if implemented)
 - [] Verify all links in emails work correctly
 - [] Verify emails are mobile-responsive
-

G. Performance & UI Testing

Page Load Times

- [] Measure homepage load time (should be < 2 seconds)
- [] Measure dashboard load time (should be < 3 seconds)
- [] Measure doctors list load time with 50+ doctors
- [] Measure appointments list load time with 100+ appointments
- [] Verify no significant lag during navigation

Database Performance

- [] Add 100+ doctors and verify list loads quickly
- [] Add 500+ appointments and verify filtering works smoothly
- [] Perform complex searches and verify response time
- [] Check if pagination improves performance for large datasets

Browser Compatibility

- [] Test on Google Chrome (latest version)
- [] Test on Mozilla Firefox (latest version)

- [] Test on Microsoft Edge (latest version)
- [] Test on Safari (macOS/iOS)
- [] Verify consistent appearance across browsers
- [] Verify all features work on each browser

UI/UX Quality

- [] Verify consistent color scheme throughout application
- [] Verify consistent font styles and sizes
- [] Verify proper spacing and alignment
- [] Verify all buttons have hover effects
- [] Verify all forms have clear labels
- [] Verify error messages are user-friendly
- [] Verify success messages are prominent
- [] Verify loading spinners appear during operations
- [] Verify navigation is intuitive and logical
- [] Verify accessibility (keyboard navigation, screen reader support if implemented)

JavaScript Console

- [] Open browser developer tools (F12)
 - [] Navigate through all pages
 - [] Check console for JavaScript errors
 - [] Verify no 404 errors for CSS/JS files
 - [] Verify no CORS errors
 - [] Verify no deprecation warnings
-

Final Testing Summary

After completing all tests, fill out this summary:

Total Tests Conducted

- [] Admin Portal: _ / __ passed
- [] Doctor Portal: _ / __ passed
- [] Patient Portal: _ / __ passed
- [] API Endpoints: _ / __ passed
- [] Edge Cases: _ / __ passed
- [] Email Tests: _ / __ passed
- [] Performance Tests: _ / __ passed

Known Issues

List any issues found during testing:

- 1.
- 2.
- 3.

Critical Bugs (Must Fix Before Submission)

- [] None found
- [] Bug 1: _____
- [] Bug 2: _____

Minor Issues (Optional to Fix)

- [] Issue 1: _____
- [] Issue 2: _____

Testing Completion

- [] All core features tested and working
 - [] All API endpoints tested and working
 - [] No critical bugs remaining
 - [] Application is ready for submission
-

Testing Tips

1. **Test Methodically:** Complete one section at a time rather than jumping around
 2. **Document Issues:** Note down any bugs immediately with steps to reproduce
 3. **Use Multiple Browsers:** Don't just test on one browser
 4. **Test with Real Data:** Use realistic names, emails, and data
 5. **Test Edge Cases:** Don't just test the happy path
 6. **Clear Browser Cache:** Clear cache between major tests to avoid cached data issues
 7. **Use Incognito Mode:** Test in incognito/private mode to simulate fresh user sessions
 8. **Test with Friends:** Have someone else test your application for unbiased feedback
 9. **Record Test Results:** Keep a log of what worked and what didn't
 10. **Retest After Fixes:** After fixing bugs, retest the affected features
-

Need Help?

If you encounter issues during testing:

- Refer to `TROUBLESHOOTING.md` for common problems and solutions
 - Check the application logs in the terminal where Flask is running
 - Review the browser console for JavaScript errors
 - Check the database file (`hospital.db`) for data integrity
-

Testing completed on: _____ (Date)

Tested by: _____ (Your Name/Roll Number)

Ready for submission: [] Yes [] No

Good luck with your testing! ✎✓