**Project Design Phase-II**

**Solution Requirements (Functional & Non-functional)**

|  |  |
| --- | --- |
| Date | 26-11-2025 |
| Team ID |  |
| Project Name | DocSpot |
| Maximum Marks | 4 Marks |

## **Functional Requirements – DocSpot**

|  |  |  |
| --- | --- | --- |
| **FR No.** | **Functional Requirement (Epic)** | **Sub Requirement (Story / Sub‑Task)** |
| FR‑1 | User Authentication | Sign up, Login, Logout for patients, doctors, and admins using email and password. |
|  |  | Password reset / change password for all users. |
|  |  | JWT‑based session handling with role stored in token. |
| FR‑2 | Doctor Browsing & Discovery | Search doctors by specialization, location, and availability. |
|  |  | View detailed doctor profile with fees, experience, and ratings. |
|  |  | Filter and sort doctors (e.g., by rating, fees, distance). |
| FR‑3 | Appointment Scheduling & Management | Patients book appointments with selected doctor, date, and time slot. |
|  |  | Patients can reschedule or cancel their own appointments (rule‑based). |
|  |  | Doctors view daily schedule and accept, reject, or mark appointments as completed. |
| FR‑4 | Secure Document Handling | Patients upload medical reports linked to specific appointments. |
|  |  | Doctors upload prescriptions/notes for completed appointments. |
|  |  | Users can view/download only their own documents with proper access control. |
| FR‑5 | Profile & Dashboard Management | Patients manage personal details and see upcoming/past appointments. |
|  |  | Doctors manage profile, consultation fees, and available time slots. |
|  |  | Admin views platform statistics (counts of users, doctors, appointments). |
| FR‑6 | Admin & Verification | Admin approves/rejects doctor registrations after document review. |
|  |  | Admin can deactivate suspicious users/doctors and monitor system logs. |
| FR‑7 | Notifications & Real‑Time Updates | Send email / in‑app notifications for appointment booking, updates, and cancellations. |
|  |  | Real‑time status updates of appointments and slot availability on dashboards. |

## **Non‑Functional Requirements – DocSpot**

|  |  |  |
| --- | --- | --- |
| **NFR No.** | **Non‑Functional Requirement** | **Description** |
| NFR‑1 | Usability | The app should provide a clean, intuitive UI so patients, doctors, and admins can complete tasks with minimal steps. |
| NFR‑2 | Security | Protect all sensitive data with HTTPS, hashed passwords (bcrypt), JWT auth, and role‑based access control. |
| NFR‑3 | Reliability | Appointment booking, cancellations, and document access should work correctly under normal and peak loads. |
| NFR‑4 | Performance | Key pages (login, doctor list, dashboard) should load within about 2–3 seconds on typical connections. |
| NFR‑5 | Availability | The system should target at least 99% uptime for core booking and viewing features. |
| NFR‑6 | Scalability | The architecture should support growing numbers of users and appointments by scaling the Node.js and MongoDB tiers. |
| NFR‑7 | Compliance & Privacy | Patient data must be stored and transmitted following healthcare privacy principles and local regulations. |