Requirements Document

User- level requirements are written from the user role's perspective. Users can be patient themselves or third-person on behalf of patient. Named information is shown quoted in these requirement statements. Information used in the user-level requirements is described in the "Common Information" section below:-'

A functional requirement describes what a software system should do. What must the system do to support the user roles? Information referenced in the functional requirements is described in the "common information" section below.

Column Header Key: BR= Business Rule Identifier, CI= Common Information Identifier, ST= Status.

Status Column Key: A= Accepted, C= Changed since last review, N (or Blank)= New since last review.

Functional Requirements

User and functional requirement statements

ID	User and Functional Requirements	BR	CI	ST
B1	The application should facilitate all the hospital functionalities.	A		
User Role	**Patient**			
U1.1	**View Registration Page of patient**			**A**
U1.1F1	The system should allow the user to find the sign up form.			A
U1.1F2	The system should allow user to fill and submit sign up form.			A
U1.2	Login to the app.			A
U1.2F1	The system should allow user to login with his username and password.			A
Goal U2	**Reserve doctor appointment**			A
U2.2	**View doctor information**			A
U2.1F1	The system should allow patient to search doctor's information.			A
U2.2	Book appointment online with Diagnostic center.			A
U2.2F1	The system should allow the patient to search and view appointment details such as specific time slot.			A
U2.2F2	The system should give confirmation for the appointment and calculate the waiting time on doctor's appointment.			A
Goal U3	**Confirm Appointment online**			A

U3.1	Search and View doctor's qualifications.			A
U3.1F1	The system should allow user to search qualifications by filters.			A
U3.1F2	The system should allow user to view different doctors online.			A
User Role	**Doctor**			
Goal U1	**Registration**			
U1.1	Registration of Doctor on the application.			A
U1.1F1	Register Online with the profile in sync with registration number from government.			A
U1.1F2	Login into the application if already signed up.			A
Goal U2	**Signup on the application with credentials**			
U2.1F1	Create profile			A
Goal U3	**Manage appointments**			
U3.1F1	View and manage appointments.			A
Goal U4	**Check test reports of patients**			
U4.1F1	Access test results of patient from Diagnostic center.			A
U4.1F2	Cancel/update appointments			A
U4.1F3	Log Out from the application			A
User Role	**Diagnostic Center**			A
Goal U1	**View and search patients online**			A
U1.1	Search the patients.			A
U1.1F1	Search the patients associated with a particular doctor.			A
U1.1F2	Register the diagnostic center in respect to reference number.			A
Goal U2	**Application Login**			
U2.1	Login /update the records of patient.			A
U2.1.F1	Login into the application as Diagnostic center.			A
U2.1F2	Edit the patient's record in respect to test results.			A
Goal U3	**Diagnostic center functionality**			
U3.1	Diagnostic center functionalities			A
U3.1F1	Add contact details of Diagnostic center.			A
U3.1F2	Update contact details.			A

U3.1F3	View appointments.			A
U3.1F4	Upload reports.			A
U3.1F5	Log out from Diagnostic center page.			A

Non-Functional Requirements

Non -functional requirements focus on the qualities that must be applied to design and implement the system. These specific standards and attributes in support of the other requirements.

ID	Non Functional Requirement Statements	BR	Cl	ST
Access Security: How well in the system guarded against unauthorized access? The extent to which the system is safeguarded against deliberate and intrusive faults from internal and external sources.				
N-ACS1	-Login/Access levels: Patient, Doctor and Diagnostic center access. -Create, Read, Update, Delete levels (CRUD) Levels			A
N-ACS2	Access permissions for the application data may be changed by the system's administrator.			A
	Password requirements – length, special characters, expiry, recycling policies			A
	Inactivity timeouts – durations, actions, traceability			A
	Data Classification / System Accreditation: All Data must be protectively marked and stored / protected.			A
Availability: How dependable is the system during normal operating times? The degree to which users can depend on the system to be up (able to function) during "normal operating times".				
N-AVL1	Patient should be able to access the reports from the application.			A
N-AVL2	Application should be functional 24 hours a day through-out the year.			A
Efficiency: How fast it can be processed? How many can be processed? How does the system respond? The extent to which the software handles capacity, throughput, and response time.				
N-EFC1	The response time of application.			A
N-EFC2	Any changes in the database should be visible to all the users.			A
Integrity How accurate and authentic are the data? The degree to which the data maintained by the software system are accurate, authentic, and without companion.				

N-INT1	The reports should be processed for the patient.			A
N-INT2	The incorrect reports might lead to false medication.			A
Reliability: How immune is the system to failure? The extent to which the software system consistently performs the specified functions without failure.				
N-REL1	The acceptance rate of system failure should be low.			A
N-REL2	Acceptance threshold for down-time should be low.			A
N-REL3	Mean time to recovery should be less- if system gets broken, how much time is required to get it running again.			A
Survivability: How resilient is the system from failure? The extent to which the software system consistently performs the specified functions without failure.				
N-SRV1	If any data is missing, the record should get flagged.			A
N-SRV2	The Application's database should be unscathed in case of any downtime/failure.			A
Usability: How easy is to learn and operate the system? The ease to which the patient/ user should be able to learn and operate with the application.				
N-USE1	The patient/user should be able to perform the necessary functions using the application.			A
N-USE2	The application should generate the report in as pdf and word document.			A
Maintainability: How easy is to upkeep and repair the system? The ease with which faults in application/ system can be found and detected?				
N-MNT1	The maintenance of the system should be performed at regular intervals.			A
N-MNT2	The customers should not be affected by the maintenance activities.			A
Scalability: How easy is it to expand the capabilities? The degree to which the system is able to expand its processing capabilities upward and outward to support business growth.				
N-SCL1	The system should be made in such a way that it can be scaled to a bigger view.			
Verifiability: How easy is it to show the system performing its functions? The extent to which tests, analysis, and demonstration are needed to prove that the system will function as intended				
N-VER1	The system functionality should be tested with test cases including all the possibilities for failure.			A

N-VER2	Unit test cases should ensure 100% branch coverage.			A
Interoperability: describes the extent to which systems and devices can exchange data, and interpret that shared data. For two systems to be interoperable, they must be able to exchange data and subsequently present that data such that it can be understood by a user.				
N-IOP1	The application should be available on android and IOS.			A
N-IOP2	The data security should be maintained: Patients data should be hidden under his account and shouldn't be available			A
Portability: How easy is it to transport? The ease with which a software system can be transferred from its current hardware or software environment to another.				
N-POR1	The system should be able to run on tablets and smartphones.			A
N-POR2	The system implementation should not be specific to any operating system.			A
Reusability: How easy is it to convert for use in another system? The extent to which a portion of the software system can be converted for use in another.				
N-REU1	Codes and backend should have functions which can be reused.			A
N-REU2	Some test cases should be in a format which can be used in future application.			A