Non-Functional Requirements (NFRs) for the System

Performance

Taking into consideration the system will support a government hospital, the system must be performant however the expectations are limited due to the lower funding in Egypt. This can be expressed as requiring the following from out system:

- Support up to 1500 concurrent users without noticeable delays.
- Critical operations (e.g., updating records) must respond within 100-300 millisecond.
- Generate reports within 1-3 seconds.
- Handle up to 200 transactions per second using workers and thread.
- Stress tested to ensure it can function without noticeable slowdown at 2x-3x the normal load for up to 15 minutes.

Scalability

In case some clinics want to join the system and work under the hospital (most hospitals in Cairo have an external clinics section) we need our system to be able to quickly scale.

- Allow scaling to support up to 5000 users as the system expands.
- Easy to integrate with: Enable data sharing between hospitals in standard formats (JSON, CSV, TSV).
- Easily accessible and secure API for the data transfer.

Availability

- Ensure 99.5% uptime, with minimal maintenance downtime, basic redundancy should be used such as 2 nodes at least.
- Keep backup servers for critical functions, can be slower than main one.
- Keep constant backups

Security

- Encrypt all data using stored and transmitted.
- Use role-based access for restricting access for certain people in the hospital.
- Comply with Egyptian healthcare privacy laws.
- Should be tested against and resilient to OWASP top 10 vulnerabilities.

Usability

We expect our patient and staff to be familiar with smartphones to a certain extent, however we have no expectations from them in terms of computers as they are not as common in smaller cities.

- Hospital user interface should be intuitive, requiring no more than 10 hours of training.
- Learnability: Patients should be able to log health metrics (e.g., blood pressure) or view their data via a mobile app with no prior training, only a 10 minute guide.
- Prioritize accessibility for less tech-savvy users.

Reliability

- Perform daily backups and store them offsite.
- Implement a disaster recovery plan to prevent data loss and make sure the backup server is constantly up to date.