Gathering Quality NFRs

1. Access and Security

Who needs to access the application?

How will they access it and what are they entitled to do within the application?

2. Availability and Reliability

When will users need to access the application? During working hours, on working days, or 24/7?

How critical is the application's functionality to the business, and how does any unplanned downtime affect day-to-day business?

3. Reliability

Will there be significant adverse negative impacts if users are unable to log in at a certain time or for a given time period?

What is the accepted probability of failure for the application?

4. Performance

Which functional requirements need fast response times and how fast should they be?

Can the application handle multiple users logging in and performing activities at the same time?

5. Usability

Is the interface easy to learn and use? Is it designed intuitively such that users can figure out what they need to do to achieve their objectives with the least number of clicks or inputs necessary? *(This question typically needs to be answered by the users.)*

Does the design adapt to the device or size of the screen being used?

Are there feedback mechanisms that prompt users on what to do next?

Have users been given a usability questionnaire, and if so, what were the results?

6. Integrity of Data

How do you ensure that redundant or inaccurate data are effectively managed?

How is data modeling related to the organization’s data policies?

How do you capture and address incomplete data?

7. Maintainability

How uncomplicated is it to keep the application operational?

Are changes straightforward to make? If something is not working, can a fix be applied with relatively little effort?

8. Scalability

Can your application adapt to an ever-increasing (or decreasing) workload? Can it accommodate more users? More functionality? More concurrent processes?

Can it work on different architectures?

9. Reusability

Does your app contain functionality that can be parceled and shared with others, or in other applications on the same platform?

Does the application make use of common functionality? Does something already exist to solve the need? Does the app need to build it if it exists in another system?

10. Interoperability

How well does the application interact and share data and information with its internal and external environments?

How reliable and easy to maintain is this process?

11. Recoverability

Is there a plan to recover the application’s data and be operational again in the case of a disastrous event?

How quickly and easily can the application’s damaged elements be recovered? Is there a backup of the information?

Source: *Citizen Development: The Handbook for Creators and Changemakers*