

Non-Functional Requirements - Checklist

Security

- Login requirements - access levels, CRUD levels
- Password requirements - length, special characters, expiry, recycling policies
- Inactivity timeouts – durations, actions

Audit

- Audited elements – what business elements will be audited?
- Audited fields – which data fields will be audited?
- Audit file characteristics - before image, after image, user and time stamp, etc

Performance

- Response times - application loading, screen open and refresh times, etc
- Processing times – functions, calculations, imports, exports
- Query and Reporting times – initial loads and subsequent loads

Capacity

- Throughput – how many transactions per hour does the system need to be able to handle?
- Storage – how much data does the system need to be able to store?
- Year-on-year growth requirements

Availability

- Hours of operation – when is it available? Consider weekends, holidays, maintenance times, etc
- Locations of operation – where should it be available from, what are the connection requirements?

Reliability

- Mean Time Between Failures – What is the acceptable threshold for down-time? e.g. one a year, 4,000 hours
- Mean Time To Recovery – if broken, how much time is available to get the system back up again?

Integrity

- Fault trapping (I/O) – how to handle electronic interface failures, etc
- Bad data trapping - data imports, flag-and-continue or stop the import policies, etc
- Data integrity – referential integrity in database tables and interfaces
- Image compression and decompression standards

Recovery

- Recovery process – how do recoveries work, what is the process?
- Recovery time scales – how quickly should a recovery take to perform?
- Backup frequencies – how often is the transaction data, set-up data, and system (code) backed-up?
- Backup generations - what are the requirements for restoring to previous instance(s)?

Compatibility

- Compatibility with shared applications – What other systems does it need to talk to?
- Compatibility with 3rd party applications – What other systems does it have to live with amicably?
- Compatibility on different operating systems – What does it have to be able to run on?
- Compatibility on different platforms – What are the hardware platforms it needs to work on?

Maintainability

- Conformance to architecture standards – What are the standards it needs to conform to or have exclusions from?
- Conformance to design standards – What design standards must be adhered to or exclusions created?
- Conformance to coding standards – What coding standards must be adhered to or exclusions created?

Usability

- Look and feel standards - screen element density, layout and flow, colours, UI metaphors, keyboard shortcuts
- Internationalization / localization requirements – languages, spellings, keyboards, paper sizes, etc

Documentation

- Required documentation items and audiences for each item