A detailed discussion of the likely non-functional requirements of the proposed system.

(list 10 non-functional requirements with its explanation at least for this task, e.g. scalability, reliability, availability, capacity, security, etc.)

Non-functional requirements

|  |  |
| --- | --- |
| System | Can the system handle updates and be up and running or does it halt, can it update within a reasonable time frame or out of hours. Also that the system is easy to use and is user friendly and simple to navigate for users. |
| Security | The system has to be secure and have a strong log in system and its only available to the required people using it and cannot be compromised. Has data encryption or firewalls to help keep it secure. Data is secure, the log in, the id numbers etc  The application must be able to avoid malevolent incidents and events outside of the system planned design. |
| Scalability | This app can be expanded to add other functions in the future and capacity for more rooms and features as the firms requirements can grow for next version updates.  So it should be easy to scale in terms of office expansion and recruitment and adding more extensions to the app. |
| Reliability | The system is reliable, it is accurate in portraying the correct data and the available user space within the office, and only the manager and CEO have access to see all bookings. The system can last over years as a worthwhile investment being cost effective and reliable with having uptime of 99.9% or an agreed service level agreement of availability. |
| Resource constrains | Processor speed , memory, disk space and network bandwidth are all non-functional requirements that need to be accounted for to make sure the system has enough resources and that it stays maintainable and reliable. |
| Usability | How easily the system can be learnt and used and taught to others. How quickly staff can pick up completing tasks & how efficiently they can be completed. The system needs to be easy to use, fast, efficient and accessible. For example:  For example users should be prompted to fill certain fields & prompted to invalid inputs & null fields. |
| Recoverability | System back up when unexpected conditions arise for example system crash |
| Maintainability | Being cost effective and easy to maintain, than having physical employees booking rooms and creating issues of overbooking and human error. The system is easier to maintain as it can keep track of bookings and showcase what is available limiting the risk of over bookings which is not possible. |
| Speed | How fast the app responds to commands when booking and cancelling is important to make sure users are not waiting an excess amount of time. That the process should not take longer than 120 seconds to make a hot desk booking. Searching availability of desk would take 30 seconds or less. |
| Load | The system is fit to expand and can handle an increase volume/load of users at the same time without comprising on speed and time without putting strain on the system it should be able to handle without crashing. |
| Interoperability | The booking system must be able to interoperate with other internal systems for example outlook etc it needs to have compatibility with shared applicates and other systems it may need to be integrated with. |
| Performance | Responsiveness of the application. Responsiveness is important as it effects scalability. Enhancing an applications performance can also improve scalability by diminishing shared resources. The booking system needs to be designed to support specific response time requirements of tasks. |
| Availability | The system performs tasks must be performed within normal times frames without malfunction. The system must have maximum up time, be available for use 7 days per week 52 weeks per year. The system must rapidly recover from faults. For this booking system we must consider application availability during weekends, holidays, system maintenance & failures. |