**Non-Functional Requirements (System Quality Attributes):**

|  |  |  |  |
| --- | --- | --- | --- |
| Classification | Requirements | Explanation | Conditions or risk to complete any requirement (tradeoffs) |
| User Usage | Usability | Measure of how well users can make use of a system's features. | NA |
| Accessibility | Accessible to all kind of users including disabled | Using CSS and html to ensure all users read access |
| Reusability | Possibility of reducing production time | Sections of class and modules to reuse |
| Modifiability | Ease adoption of all future components and modifications | NA |
| Interoperability | Efficient information transformation | Structured database creation to remove redundant data |
| Customizability | Allow user required experience | Using html tags user’s preference is updated (e.g. Celsius or Fahrenheit) |
| Development | Maintainability | Time and speed to restore system during failure/ faults occurring | Using scalable and component base architecture and modules |
| Manageability | effectively manage a system to build, integrate and test code | Using Travis CI and GitHub |
| Testability | Verifying functionality/feature result according to requirement | Based on use cases the feature results will be ensure |
| Extensibility | Ability to add new functionality | NA |
| Operation | Performance | Response per request/task, utilization | 1sec |
| Scalability | Improve latency, throughput | Using efficient REST API interface |
| Availability | Time of system running and to repair faults | All features available to use throughout week 24\*7 |
| Reliability | Efficient performance without failure for specific users/time | Users should access features 98% of the time |
| Recoverability | Ability to recover from system failure | All features available within 1 day of unplanned downtime/ system crash |
| Security | Auditability | Users evaluation/authenticity | NA |
| Integrity/confidentiality | Non-disclosure of data to unauthorized uses | User login or signup required to access data |
| Other |  |  |  |

**Performance Requirements:**

|  |  |  |
| --- | --- | --- |
| Attributes | Expectations | Constraints |
| Application Throughput, response time per transaction | < 2 seconds | REST API interaction between two system |
| Application Utilization rate | 75% | NA |
| System Response time to establish server connection | < 2minutes | NA |
| Time to end session due user inactivity | 4 minutes | NA |
| User profile logout time during inactive | 8 minutes | NA |
| System Utilization rate | 60% | NA |
|  |  |  |

**Business Requirements:**

1. The Portal app should comply with all legal government regulations so that potential business activities are not affected by lawsuits and/or government interference.
2. The application should be copyright-registered so that rivals cannot copy our work. To prevent copyright lawsuits, however, the application should not be identical to an existing application.
3. Profit margins should rise for a certain percent per year.
4. Updates must be published on a regular basis to keep the application up to date and mitigate threats such as security breaches, which can lead to the theft of personal information such as disclosure of user personal information and payment details.
5. The app should be more appealing to potential users, by providing ads on social media.