**Performance Requirements:**

1. The system response time shall be no more than 0.5s under normal conditions (users<100000)
   1. The execution time shall be no more than 0.5s under normal circumstances (users<100000)
   2. In case in under heavy traffic (usage of more than 100000 users at once ) response rate and execution time shall be no more than 1s
   3. Under heavy backlog of complaints response rate and execution time for emergency complaints shall remain same (0.5s)
2. The system is available 100% for the user and is used 24 hrs a day and 365 days a year.
   1. The system shall be operational 24 hours a day and 7 days a week.
3. The system shall provide 100% access reliability.
   1. The system shall accurately provide real-time information taking into consideration and showing correct information at all times

**Security Requirements:**

1. The software programme will only be accessible to authorized users who have a valid login and password.
2. New LESCO customers can use their meter number to register or sign up for the system.
   1. The modules and functionalities that users have access to or are available to the users (Employee and Customer) will differ depending on their responsibilities.
   2. Users will be able to connect into the programme using their username and password and access the modules or functionalities that are appropriate for their roles.
3. To prevent automated login, CAPTCHA will be used. If there are any issues while accessing the system, error messages will be presented.
4. The software is to follow SIL 3 safety certification
5. The system shall be able to block malicious or unauthorized actions that could potentially destroy or hack the system.

**Modularity requirements:**

1. System shall follow Micro services architecture
2. System shall implement cloud computing with vertical scaling
3. System shall implement CDN(content delivery network) from cloud flare for front end
4. System shall make sure that software can be installed into the common, expected locations on the system
5. System shall first be divided into two components Common Utility Components and Frameworks and Infrastructure Common Components
6. System shall further decompose into further components based on the two components discussed above by feature and layer
7. System implementation of a module is private. Their data structures, functional structure, and dependencies should not be exposed
8. A module’s implementation should be flexible enough that any change can be made without affecting any of the module’s users
9. System shall have a central database and two more for replication purposes
10. System shall integrate already available payment module (Everyware)

**Reliability requirements:**

1. The system shall use the following metrics to measure reliability

* Probability of failure on demand (POFOD)
* Rate of occurrence of failures (ROCOF)
* Availability (AVAL)

1. The system shall uses estimation modelling to determine reliability
2. The system shall implement Self-monitoring to detect inconsistencies
3. The system shall provide a Provide a handler for all exceptions
4. The system shall minimize the use of error-prone constructs
5. The system shall provide restart capabilities
6. The system shall implement timeouts when calling external components
7. System shall also implement cache configuration to avoid database overloading