* Our interface should be easy to understand even for those who are not tech savvy

### The ISO's taxonomy of requirements

2. Performance efficiency

* Our system should have fast performance to ensure the frustration of use does not discourage students from logging in their contributions correctly.
* Our system should ensure that performance and speed are not compromised during a peak usage time, to ensure that student do not lose progress of their contributions during these times.

3. Compatibility

* Our system should be able to access any information it needs / has legal access to from the device it is running on.
* Our system should not slow down / interfere with other programs running on the user’s computer, as we need to make sure that our software can run seamlessly in the background as the user is doing their contribution work.

4. Usability

* The system should be self explanatory, so the users understand what they are looking at
* The system should follow easy flow of work, so new users can learn to perform different tasks easily
* The error messages generated in any step of using the system should be self explanatory and directing the users to the next step
* The system should have a nice and pleasant interface
* The system should be accessible to different users

5.Reliability

* Our system should have a specific way to recover from data restoring itself in case of system failure.

6.Security

* The system should prevent unauthorized people from accessing or modifying code or data

7.Maintainability

* The code structure of our system should be broken down into logical chunks so that if fixing/upgrading is required it is easy to navigate to the appropriate file to make the changes.
* Our system should be made to be easily testable for errors
* Dependencies should be reduced as much as possible to ensure changes to one part of the system does not break other parts of the system
* Our system should give feedback when the system breaks, so that developers and users can know what went wrong when our system fails.

8.Portability

* Our system should be easily accessible / downloadable so that it does not become a burden for the users that are using it
* Our software need to be an improved version of other marking softwares that currently exist
* Our software needs to maintain its usability to a high standard across different devices and operating systems. This is because students and markers use a variety of different types of devices to do their work

**How will implementations be verified?**

Make sure that the design implementations are implemented according to Daniel Norman’s and Scneiderman’s design principles and that they fulfill the user’s requirements through user focus group testing?