## Non-Functional Requirements

### Security requirement

* A user will be required to create a user account.
* On creating an account:

1. The user will be asked to register with a Google account.
2. Should the user forget their password they will be required to answer a security question, under password rest, in order to gain access.
3. The user will have three attempts to enter a correct password, Account will be locked upon three incorrect attempts and will need to utilise password reset to gain access.
4. User login details will be authenticated against the User database, which will be a stand-alone database to add an additional layer of security.

* The platform will be hosted on IBM’s Cloud which provides its own security.
* The platform should restrict activity outside of the user’s role.
* The platform should log out after a period of inactivity.

### Maintainability requirement

* The User should not suffer any interruption in usability while updates are pushed to the server. In effect the user should only notice changes on next login.
* The user should have the ability to report bugs and feature requests through a support portal.
* Each update should have required regression testing performed.
* Each update to the platform should be added to the release notes available on the site.

### Robustness requirement

* The Platform should return to previously saved point upon return from outage.
* Upload validation - An error should be returned if:

1. The file format is not supported.
2. The file size is to large.

* Input validation – An error should be returned if:

1. The user exceeds maximum word count.
2. The password criteria have not been met.

### Portability requirement

* The platform should conform to HTML standards as defined by W3C.
* The platform should be available across multiple devices including PC, Tablet, with various operating systems.
* The platform should support internet Explorer, Opera and Safari but shall be optimised for Chrome.
* The platform should be light-weight to enable use at slower internet speeds.

### Reusability requirement

* The user should have the ability to save a “Kanvass”, to use at a later date. This will be available in the history section of the “Kanvass”.
* The user should also be able to share on multiple platforms (Facebook, Twitter, email) the “Kanvass” for use on several devices.
* The platform should have a uniform look and feel between pages.
* The platform should make use of toolbars and icons.