

<u>Artifact #</u>	<u>Artifact</u>	<u>Version</u>	<u>Status</u>	<u>Priority</u>	<u>Description</u>
Feature					
1	5 Predictive Text	1.0	Submitted	Not required	When the user types part of a word and pauses, the editor should suggest possible completions for that word. These should be based on the names of variables, classes, packages and functions to which the user may be referring. Rational: This can speed up the development process dramatically as users don't have to look up function names and in some cases can avoid typographical errors caused by typing the whole of a function name. Source: Bob Ray, Cardiff Office.
2	4 Text Hi-Lighting	1.0	Submitted	Not required	The editor should highlight text in a way which is meaningful to the user. It should recognize keywords, constants variables etc. and display them in a chosen colour. The editor should highlight text in a way which is meaningful to the user. It should recognize keywords, constants variables etc. and display them in a chosen colour. Where no user preference exists a default colour may be chosen. Rational: Its much easier for users to read code if it is appropriately coloured (highlighted). Source: Bob Ray, Cardiff Office.
3	3 Auto Save	1.0	Submitted	Important	The editor should store changes in a temporary file such that latest version can be restored in the event of a crash Rational: The editor needs to be able to recover files to be effective. This maintains user confidence in the software and reduces the chances of major productivity loss though systems failuers.
4	2 Editing Files	1.0	Submitted	Must have	Source: Ray Wilson, Glasgow Office The editor should provide a feature to edit open files of varioius types. Rational: For the editor to be functional and effective the editor needs to be able to edit files.
5	1 Opening Files	1.0	Submitted	Must have	Source: Ray Wilson, Glasgow Office The editor should provide features to allow the opening of files. The files can be of various types Ascii, extended Ascii and UNICODE. Rational: For the editor to be functional and effective the editor needs to be able to open files. Source: Ray Wilson, Glasgow Office
Requirement					
1	9 Opening Files	1.0	Submitted	High	User must be able to open an existing file. The contents of the file should be read into a buffer for editing.
2	8 Text Display	1.0	Submitted	Medium	The text from the buffer should be displayed in a window, this window should be placed centrally and provide editing facilities.
3	7 User Save As	1.0	Submitted	Medium	When the user issues a Save As command, either though shortcut keys or a menu option the buffer containing the file should be written to a new file of the users choosing. The users must be asked for the name and location of the new file.
4	6 User Save	1.0	Submitted	High	When the user issues a save command, either though shortcut keys or a menu option the buffer containing the file should be written to its original file location.
5	5 Maintanance of a	1.0	Submitted	Medium	The edit buffer should write the content of the file to a temporary file at user defined intervals.
If the file cannot be written then the system should warn the user that a temporary file could not be created and that no provision is made for system failure. If the system fails it should check the temporary file, if the file exists (user has not saved) it should load this file first giving the user the opportunity to save.					

<u>Artifact #</u>	<u>Artifact</u>	<u>Version</u>	<u>Status</u>	<u>Priority</u>	<u>Description</u>
					If the user has saved the file the temporary file can be deleted.
6	4 Buffer Editing	1.0	Submitted	High	A buffer should be used to edit the contents of a loaded file. The buffer created by loading should be parsed to create a library for the predictive text based on rules on the operators of the language store. When loading a file the system should identify the type of file being loaded and ensure that the system informs the user as any lack of capabilities in terms of text highlighting etc. When a new file is created the user should be given an empty buffer to edit.
7	3 Generation of Predictive	1.0	Submitted	Low	
8	2 Identify file type of file to be	1.0	Submitted	High	
9	1 Must be able to create new	1.0	Submitted	High	