



Association of Maritime Managers in
Information Technology & Communications



**MARITIME SOFTWARE
QUALITY GUIDELINES**

AMMITEC - Maritime Software Quality Guidelines

Version 1.0 September 2017

Terms of Use

The information contained in the Maritime Software Quality Guidelines document is for general guidance purposes only. The information is provided by AMMITEC and while we endeavour to keep the information up to date and correct, we make no representations or warranties of any kind, express or implied, about the completeness, accuracy, reliability or suitability with respect to information contained in this document. In no event will we be liable for any loss or damage including without limitation, indirect or consequential loss or damage, or any loss or damage whatsoever arising from loss of data or profits arising out of, or in connection with, the use of this document. Any reliance you place on such information is therefore strictly at your own risk.

Table of Contents

AMMITEC - Maritime Software Quality Guidelines	2
1. Maritime Software Quality Guidelines	5
An Introduction	5
2. Fifteen Things to Ask from Software Vendors.....	6
1. Users' Forum.....	6
2. Functionality and Correctness	6
3. Help.....	6
4. Usability and Accessibility.....	7
5. Effective error/bug handling	7
6. Proper Documentation/Manuals	7
7. Performance and Efficiency.....	8
8. Scalability and Reliability	8
9. Flexibility and Extensibility	8
10. Customization and Configuration.....	8
11. Installation is easy and is all inclusive.....	9
12. Version Control and Formal User Update	9
13. Security	9
14. Communications between Vendor and Client	9
15. SLA	9

1. Maritime Software Quality Guidelines

An Introduction

AMMITEC presents a set of Software Quality Guidelines in an effort to revive a decade-old discussion to persuade all stakeholders that Shipping Software Quality needs improvements.

AMMITEC is focusing primarily on the quality of the on board Software, for well-known reasons:

- a) Reporting problems and communicating a solution is not as fast, easy and efficient as on shore. Time Zone differences, Language, Weather, and various operational problems impede this effort.
- b) Crew training is limited due to lack of time and crew rotation.
- c) For the stated reasons, on board software must be PERFECT (bug-free) and have the best-possible User Interface.

This set of Guidelines is based on ISO Software Quality Standards (ISO/IEC 25051:2014), addresses the quality of “Ready to Use Software Products”, and covers most types of on board software.

AMMITEC, the Association of Maritime Managers in Information Technology and Communications www.ammitec.org, has created a team of ICT Professionals which aims in raising Maritime Software Quality. The publication of these Guidelines aims to initiate a technical discussion between Developers and Shipping Companies, which hopefully will lead to the improvement of on board software, achieving efficiency, effectiveness, and correct data.

«Any information from your side that will towards Maritime Software Quality is most welcome».

Please send your enquiries and suggestions to info@ammitec.org

Best Regards,

AMMITEC BoD

2. Fifteen Things to Ask from Software Vendors

1. Users' Forum

We request the immediate implementation of a FORUM on the developer's official site. The Forum must be at the disposal of all Registered Users and allow the dialogue between the users. It should also allow the upload of screenshots and videos, accompanied by suggestions for improvements or solutions to well documented problems.

2. Functionality and Correctness

Functionality and correctness, is the conformity of the software with actual requirements and specifications. When/if the software is tested against the specifications given by the vendor, it should be found in full agreement with them. In case of disagreement, the vendor is required to comply fully with the specifications.

3. Help

Help should be:

- a. Online (for fast access). It usually contains the manual.
- b. Context sensitive (for precise help on the topic the user is currently interested in)
- c. User Editable (allows the user to add personal notes)
- d. Should contain FAQs, pictures, diagrams & video
- e. A Tutorial (or a CBT or a Demo) should be available for each application. Absolutely necessary for Vessel Applications.

Good support is essential to a good user experience. Effective support should have the following characteristics:

- a. Continuously improve user's interface and documentation, based on customer feedback
- b. Continuously improve the FAQs List and publish them.
- c. Customers should be encouraged to look at documentation: Help, FAQs, Forum etc. before calling support.

4. Usability and Accessibility

Usability is the degree to which software can be used by specified consumers to achieve quantified objectives with effectiveness, efficiency, and satisfaction in a quantified context of use. User interfaces are the only visible parts of software according to the viewpoint of user. So, simplicity, taking less time to complete a job and fast learnability are very important. In short, principles like “Don’t Make me Think” & “Don’t Make me Click” should be the guiding principles of the user interface.

5. Effective error/bug handling

When a program comes across an error, it should make the error known using Plain English language. No jargon is allowed. The program should provide the user with a Screen Capture utility which includes the user’s description of the problem and notifies both the developer and the shore office. Recovery from Errors is an extremely important characteristic of software (especially for vessel software), and it should try it’s best to guide users to recover fast from errors without the need for human intervention.

6. Proper Documentation/Manuals

Typically, the user documentation describes each feature of the program, and assists the user in realizing these features. A good user manual should go so far as to provide thorough troubleshooting assistance. It is very important for user manuals not to be confusing, and be kept up to date. User documents need not be organized in any particular way, but it is very important for them to have a thorough index. Consistency and simplicity are also very valuable. User documentation is considered to constitute a contract specifying what the software will do. However, there are two broad ways in which user documentation can be organized.

a. Tutorial

A tutorial approach is considered the most useful for a new user, in which they are guided through each step of accomplishing particular tasks, an absolute necessity for vessel applications.

b. Task Oriented

The manual must explain how a certain Task is accomplished. The format should be «If you want to achieve “THIS” then “DO THAT” ». An Index of Tasks is also necessary. This is important especially if a Task involves a large sequence of clicks and is not part of a menu. Usually Tasks are explained during the Training, but few users can memorize them.

7. Performance and Efficiency

Performance is mostly about response time of the software. This response time should be within acceptable user limits (e.g. max. a few seconds), and should not increase if transaction load increases. Efficiency must be supported with resource utilization. Optimal source/performance ratio must be aimed.

8. Scalability and Reliability

A scalable system responds to user actions in an acceptable amount of time, even when load increases (e.g. number of users, amount of data, etc.). Reliability also stands for the integrity and consistency of the software even under high load conditions. The vendor should provide proof of this capability.

9. Flexibility and Extensibility

Flexibility is the ability of software to add/modify/remove functionality without damaging current system. Extensibility is the ability of software to add functionality without damaging system, so it may be thought as a subset of flexibility. The vendor should provide proof of this capability.

10. Customization and Configuration

Configuration is the normal set-up of the software, such as parameters/user preferences, layout and naming of fields, and workflows. They are user defined and since they do not require changes to the source code, they form a good feature of a program.

Customization, on the other hand, requires changes to the source code and usually implies deviations from the official specification in order to meet additional/different requirements of a company. The problem usually is that customizations are performed under heavy pressure and in short time limits that affect badly the quality of the code and leads to problems with configuration.

Try to reduce Customization with built-in flexible Configuration.

11. Installation is easy and is all inclusive

Installation on a vessel should be a well prepared and organized job. It should be possible to be carried out by a crew member in most cases and the installation package should include everything that is required (for example, no external libraries or DLLs should be required).

12. Version Control and Formal User Update

Users should be updated as soon as new software release is installed. A description of all the changes must be supplied and the e-manuals must be updated asap. A detailed Log of changes is kept by the developer and the IT Manager. All upgrades should be first implemented in a test environment, tested and once accepted, migrated to the “live” environment.

13. Security

The data exchanged via email or other non-encrypted way between Vessel and Office, must be encrypted. Direct access to the Database must obey the Security Laws of each company. Data exchanged via a VPN channel, is considered secure at the moment. The software must meet the password complexity best practices, i.e. password complexity, password frequent changes etc.

14. Communications between Vendor and Client

Clear-cut instructions should be set by the Vendor describing the methods of communications whenever any support is needed (i.e. points of contacts for all modules from both the client and the vendor). Set procedures should be followed from both parties else control and monitoring will be lost, as anyone will be able to contact whoever they believe is appropriate. The Vendor should also comply with client’s communications policies (e.g. who is allowed to submit tickets, communication language should be that of the Client, etc.).

15. SLA

SLAs should be clear and penalties should be stated when commitments are not met. **Contracts:** Pricing and support agreement policies should be very clear. Special attention should be given to pricing schemes during systems updates and versions upgrades.