Sample Marketing Requirements Document for PRODUCT X
Author:

Note: this is a sample MRD from a now-obsolete model on a product that no longer exists. All elements are to be updated to current technical and marketing standards.

1.0 Scope

This is the Marketing Requirements Document for **PRODUCT X** Version 2.0. It comprises requirements for the entire first release, including maintenance upgrades, multiple license packages and upgrade versions.

The purpose of PRODUCT X will be to …..

2.0 Product Release Strategy

The product will be released in _____.

It will be released through The Company's standard channels of distribution world-wide. French, UK and German versions will be released simultaneous to the US version. At the time of the release, the following versions will need to be completed:

> PRODUCT X Version 4 3.5" version
> PRODUCT X Version 4 CD version
> PRODUCT X Downloadable Edition

Upgrades
(Policy on upgrades from previous versions)

2.1 Release 2.0 Overview

Product: PRODUCT X Version 2.0

Current Market Conditions

The nature of enhancements in PRODUCT X version 2

Timeframe constraints

Currently unresolved issues.

3.1 Functional Requirements

Priorities of requirement.

- Priority 0 -- absolutely must be in release.
- Priority 1 -- Highly desirable, should consider removing from release only if amount of effort required is not consistent with timeframe requirements.
- Priority 2 -- Desirable for this release, should include if time and resources permit.

1.   **(Item)**
        Priority: 0
        Description

22.   **Etc.**

Other Changes required:

3.2 Software Development Requirements
1.   The source code shall be complete and clearly commented.

2.   The source code shall be run through Lint and BoundsChecker, and must pass each product's tests.

3.   The program shall be built with commercially available tools and/or libraries.

4.   Any libraries that are not commercially available must include source code.

3.3 Environmental / Compatibility Requirements

1.   The product must support any hardware environment on which MS-Windows will run.

2.   The product must support all versions of DOS on which MS Windows will run, including but not limited to MS-DOS and PC-DOS v. 3.00 and later, DR-DOS 6.0 and Novel DOS 7.0.

3.   The product must be compatible with all versions of Microsoft Windows 3.1, Windows for WorkGroups 3.1, and Windows 95.

4.   The product must be compatible with all then-current versions of MS-DOS, IBM PC-DOS and Novell DOS commands and utilities.

5.   The product must be compatible with MS-Windows 32-bit file access and 32-bit disk access.

6.   The product must be compatible with Stacker, DoubleSpace, and DriveSpace disk compression utilities, in addition to any other disk compression software that may be included with or compatible with Windows 95.

7.   The product must be compatible with Program Manager, Power Desk, PC Tools Desktop, and Norton Desktop for Windows.

9.   The product must be compatible with all popular networks available at the time of release, including Banyan Vines, Novell Netware, Microsoft LAN Manager and Artisoft's Lantastic. Special attention must be paid to Windows for Workgroups.

10.   The product must be compatible with all currently-shipping Company products, including ….

11.   The product must run in Windows 3.1 Standard Mode, as well as 386 Enhanced Mode.

12.   Upgrade requirements are stated in the Product Release Strategy section above.

13.   All known common video resolutions must be supported.

14.   The Windows portion of the program must run comfortably in the lowest common denominator Windows 95 environment: 4 MBS RAM, 80 MB hard disk.

3.4 Special Quality Assurance Compatibility Requirements

3.5 Performance / Resource Requirements

1.   All functions in the program must be available via keyboard or pointing device.

2.   The program must not crash itself or the machine, and must not cause Windows General Protection Faults.

3.   The product must not fail to release Windows System Resources, except for known behavior of the Microsoft Foundation Classes.

4.   The package must multitask cooperatively with other applications. No slowdown shall be perceptible to the user.

5.   The program shall respond appropriately to inadequate memory conditions and to inadequate disk space. The product shall warn the user of the error condition, allow the user to rectify it, or will the user fail to rectify the problem, the program shall refrain from proceeding.

6.   The program will act on user requests as quickly as could be expected given the request. This will not be an unusually slow program.

3.6 Internationalisation Requirements

The product must be released simultaneously in the United States, Canada, France, Germany and the United Kingdom, and thus must be built for fast translation.

A double-byte enabled version will be required at least _____ days after the product's US release.

Internationalized products will share the same code base as the US version.

3.7 Problem Fix Requirements

1.   Before the final release of the program, all bugs marked as priority 10 through 12 in the Quality Assurance bug-tracking database will be fixed.   Bugs not fixed, (marked as deferred) must be agreed upon by all parties.

2.   Any problems detected during the beta test cycle that conflict with this document or with the Functional Specification must be resolved by release time. This resolution may include a change to the MRD or the FS.

3.   The Company shall provide problem reports immediately upon request to all outside developers. Developers within The Company will have access to the Quality Assurance Bug Tracking System.

4.   The Company shall assign priorities to bugs with 12 being the highest priority and 1 being the lowest priority. All bugs with priority of 10 through 12 shall be considered "show stoppers" and shall receive immediate attention.

5.   Problems that are fixed in a specific beta release will be marked as "fixed" by the developers on the bug report. That bug report will then be sent back to QA for verification at the same time the code with the fix is sent to QA.

6.   Beta maintenance releases will be sent to QA in no longer than two week intervals and no more often than one week intervals unless agreed upon by all parties involved.

7.   Bugs shall be fixed as soon as they are found.

3.8 User Documentation Requirements

(Detail doc changes)

A functional spec will be required at least 2 months before the documentation goes to the printer.

QA must proof the documentation before going to press.

3.90 Special Beta Release Requirements

1.   Each beta test cycle shall last not less than five weeks.

2.   There will be at least two outside beta cycles. Both of these releases will have a README file listing the known bugs, fixes from the last release, and a quick guide on how to get up and running.

3.   The help, if it is not available, will display a message that says there is no help right now, when the user hits F1 or calls upon help using a menu or button.

4.   All known GPFs will be fixed prior to shipping the first beta release to outside testers.

5.   All User Interface issues will be resolved before releasing the first official beta.

3.91 Maintenance Release Requirements

1.   No subreleases are planned at this time. As with any new product, there is likely to be a set of bug releases and maintenance fixes that address issues that we will neither recognize nor understand until the product has been in release for some time. As a matter of general policy, we would desire minor bug fixes within 60 days of the first release, and a significant upgrade release within eighteen months.

4.0 Other General Release Requirements

1.   The Install and Help files will be built into the project such that all programs function as described in the appropriate areas in this document.

2.   The package is not to be released as version 2.0 until agreed upon by all parties involved.

3.   Outside developers shall be accessible via, at a minimum, phone, fax, and e-mail during regular working hours, or at times to be mutually agreed upon by The Company.