

WIMBORNE MODEL TOWN REQUIREMENTS CAPTURE

Guidelines for 'Makers'

Table of Contents

1 INTRODUCTION.....	2
1.1 Why Capture Requirements?.....	2
1.2 Glossary of Terms.....	3
2 WRITING REQUIREMENTS.....	4
2.1 Top Level Requirements.....	4
2.2 Developing the Requirements.....	4
2.3 What Should Not be in the Requirements Specification.....	4
2.4 Developing the Project.....	4
3 CONCLUSIONS.....	4



This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-sa/4.0/>.

1 INTRODUCTION

This short document is intended to assist 'Makers' who are capturing requirements for a reasonably complex home or non-commercial Project. Examples might be the creation of a robot from scratch (not from a kit) or the 'Lighting' or 'Minster Bells' Projects at Wimborne Model Town (WMT). Such projects typically require makers to work as part of a team and to exhibit multiple skills and often include electronic hardware, computer processing, software and mechanical engineering.

The whole subject of Requirements is an essential part of professional engineering and one that more often than not has a decisive role in determining the success of any complex undertaking. What is outlined here is a sensibly distilled subset of that whole, appropriate for the needs of WMT projects.

Sufficient guidance is given to enable project teams to ensure that all of the requirements are captured and that they are unambiguous and attainable.

1.1 Why Capture Requirements?

Many hobbyists never formally capture any requirements at all and 'make things up as they go along'. This is fine as far as it goes, providing that the project is purely for hobby purposes and no third-party individuals or organisations are depending on the outcome. Where third-parties are involved they will always have 'Expectations' and these expectations may not actually be the same as the maker's, or obvious to them

So the key reasons why Requirements Capture should take place are:

1. To ensure that all aspects of the Project have been considered.
2. To ensure that all of the requirements are written in a clear, unambiguous manner that is understandable to all members of the team, including non-technical 'stakeholders'.
3. To manage Expectations. Once the Requirements have been fully reviewed and agreed by all stakeholders, then its is far less likely that at the end of the project someone will say "I thought it was going to...."
4. To provide 'hooks' for the design of the Project. As the Project develops, it is easy to see if all of the Requirements are going to be met, if the design and implementation stages of the project are traceable back to the Requirements Specification.
5. It is much easier to estimate costs for the Project if every Requirement has been followed through into the design stage.

1.2 Glossary of Terms

The following definitions apply to this document:

Derived Requirement	Requirements that emerge as a consequence of meeting a higher-level Requirement.
Desirable	This Requirement should be included if time and / or funds permit.
Essential	This Requirement has to be included.
Expectations	What any given Stakeholder believes the outcome of the Project to be.
Maker	Person who develops software, hardware or systems as a hobby.
Must	The requirement is mandatory for Safety or Legal reasons
Requirement	A single statement of what a project, or an element of a project, is to deliver.
Shall	The requirement is mandatory in order that the project may be successful.
Stakeholder	Individual or Organisation who has an interest in the outcome of a Project.
Traceability Matrix	Table or system of cross references between the system element and the Requirement that drove that part the design.

Notes:

1. The terms 'should' and 'may' are also common in Requirements engineering. For simplicity, these terms may be omitted; so only shall and must are used, coupled with an indication of how important it is to implement the requirement. If a 'Desirable' requirement is implemented, then it's importance then becomes Essential.
2. An example of a Derived Requirement is where the top-level requirement is to build a car. One Derived Requirement would that the car 'shall have wheels' etc. At a lower level still, the wheels 'shall have tyres'.
3. Stakeholders are the Management and Trustees of WMT together with the Users and the Maker(s).

2 WRITING REQUIREMENTS

2.1 Top Level Requirements

All Requirements Capture should start with identifying the aspirations of the stakeholders. This might be achieved by interviewing all interested parties, by the compilation of a paper with ideas of what the stakeholders would like or by a 'brain-storm'.

Once the Top-level Requirements are understood, they should be formally set out in a document, eg a Requirements Specification. This forms the starting point for all subsequent development.

2.2 Developing the Requirements

Starting from the Top-level Requirements, Derived Requirements are then added and reviewed. The level of detail should be increased to the point where everyone understands what is being asked for and sufficient detail is provided to allow the Maker(s) to develop the Design.

2.3 What Should Not be in the Requirements Specification

A Requirements Specification should contain only what is required, not how the requirement is to be achieved. For example the Requirement might be for a car that can do 90 mph, but not that it is necessarily made by Ford, (unless that is also a requirement, see below)). The only exception to this rule is where the stakeholders have a particular reason for needing something to be done in a particular way. So for example, they may wish the software to be written in a particular language so as to promote consistency or aid understanding. In another instance the hardware platform may be specified because that particular platform is readily available, cheap or already used in other projects.

2.4 Developing the Project

During Project development it is wise to link the solution to the Requirements using a 'Traceability Matrix'. For example in the Design document, a simple Table will prove extremely useful as a cross check that the solution will meet the requirements.

Depending on the complexity of the system, it can also be useful that the testing of the solution includes a means of confirming that each Requirement has been met.

3 CONCLUSIONS

If the guidance in this document is followed, then it is far less likely that a Project should come to a close with some cause for disappointment because the stakeholder aspirations have not been met, or the budget has been overspent.

Furthermore, as problems arise, it is much easier to discuss what should be done about the issue if the impact on the Project is fully understood with respect to these stakeholder aspirations. This works both ways. It may emerge that the problem encountered relates to a feature that the stakeholders didn't even know that they were getting.