



Requirements Management Tool rmtoo

Overview & Feature Set

by flonatel GmbH & Co. KG

Version 7

Content

- 1.The idea behind rmtoo
- 2.Features
- 3.Introduction
- 4.10 dos and 10 don'ts
- 5.Future

1.The idea behind rmtoo

Or: How to solve problems

Idea

- Requirements Management is about solving problems.
- This process is iterative: one solution may result in new problems which in turn form the basis for new solutions.
- Problem solving is mostly a *Divide et Impera*: divide the problem into smaller ones and solve them.

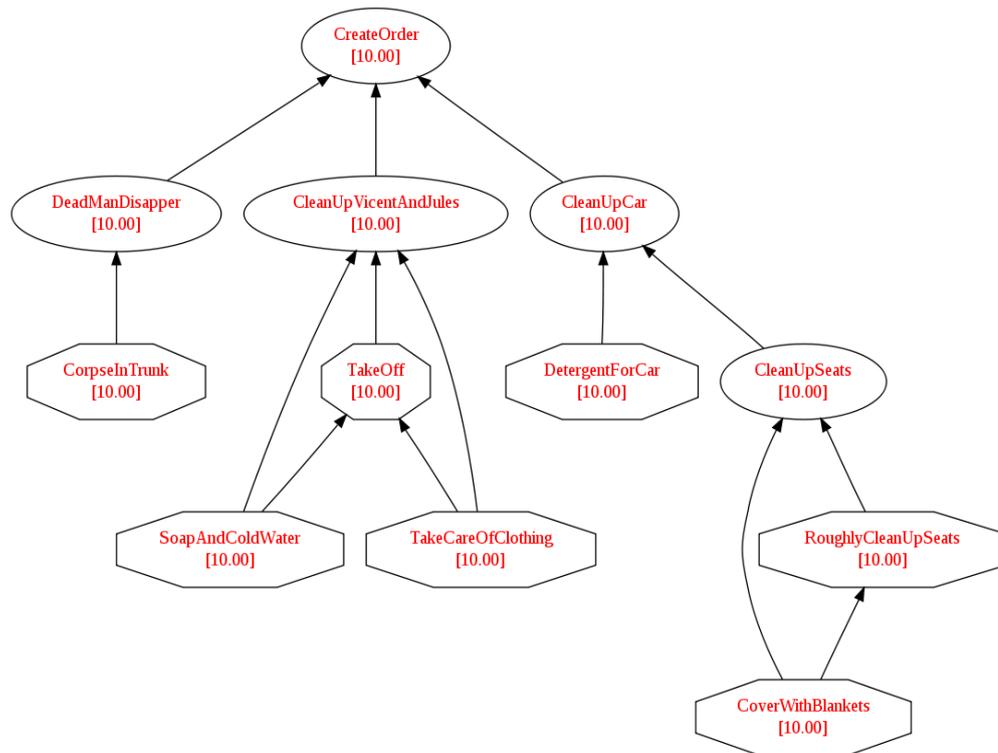
Idea: Example (1/3)

- Mr. Wulf (from the movie Pulp Fiction) solves problems.
- He divides his 'problem' into three parts:
 - **Clean up car**
 - **Clean up Vincent and Jules**
 - **Make dead man disappear**

Idea: Example (2/3)

- Then he solves these problems – one by one by dividing them up into smaller ones.
- Example: Clean up Vincent and Jules
 - take off clothes then
 - clean with soap and cold water.

Idea: Example (3/3)



- This is the problem – solution tree.
- Each problem can be seen as a requirement.
- Many solutions mean new problems – solutions are also requirements.

rmtoo

- Created for problem solving.
- It's a tool to support you. Not the other way round!

2.Features

Features (1/3)

- *rmtoo* works on plain text files.
- *rmtoo* supports version control systems.
- *rmtoo* supports a wide range of output formats.
- *rmtoo* supports a unique way of structuring requirements: topics.

Features (2/3)

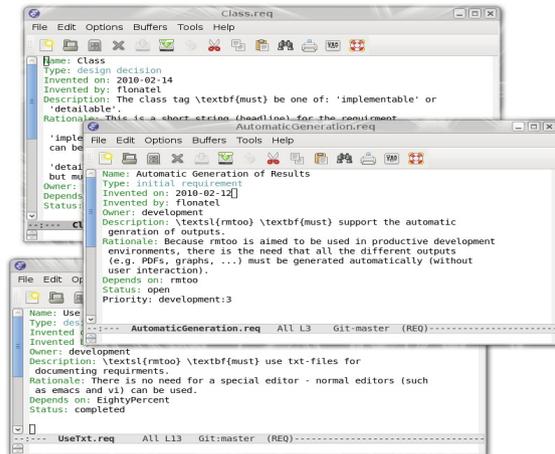
- *rmtoo* can be fully integrated into existing development environments.
- *rmtoo* checks the quality of requirements with the help of heuristics.
- *rmtoo* works on local files (you have the control).
- *rmtoo* has minimal system dependencies.

Features (3/3)

- *rmtoo* comes with a complete specification in *rmtoo* format itself (unique!).
- *rmtoo* supports you: fast and structured working is a guaranteed.
- *rmtoo* enforces you to 'Think before you write'.

3.Introduction

Overview



rmtoo



3 Initial Requirements

This is the initial - creating all the initial requirements.

3.1 rmtoo must work on Requirements

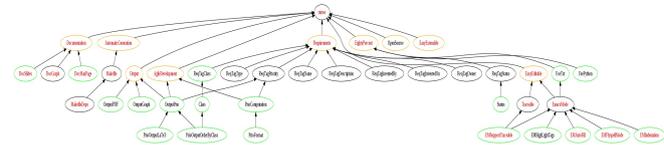
Description: rmtoo must work on requirements.
Rationale: That is what it's about.
Depends on: 2.1 rmtoo

Id	Requirements
1	Requirements
2	Requirements
3	Requirements
4	Requirements
5	Requirements
6	Requirements
7	Requirements
8	Requirements
9	Requirements
10	Requirements
11	Requirements
12	Requirements
13	Requirements
14	Requirements
15	Requirements
16	Requirements
17	Requirements
18	Requirements
19	Requirements
20	Requirements
21	Requirements
22	Requirements
23	Requirements
24	Requirements
25	Requirements
26	Requirements
27	Requirements
28	Requirements
29	Requirements
30	Requirements
31	Requirements
32	Requirements
33	Requirements
34	Requirements
35	Requirements
36	Requirements
37	Requirements
38	Requirements
39	Requirements
40	Requirements
41	Requirements
42	Requirements
43	Requirements
44	Requirements
45	Requirements
46	Requirements
47	Requirements
48	Requirements
49	Requirements
50	Requirements
51	Requirements
52	Requirements
53	Requirements
54	Requirements
55	Requirements
56	Requirements
57	Requirements
58	Requirements
59	Requirements
60	Requirements
61	Requirements
62	Requirements
63	Requirements
64	Requirements
65	Requirements
66	Requirements
67	Requirements
68	Requirements
69	Requirements
70	Requirements
71	Requirements
72	Requirements
73	Requirements
74	Requirements
75	Requirements
76	Requirements
77	Requirements
78	Requirements
79	Requirements
80	Requirements
81	Requirements
82	Requirements
83	Requirements
84	Requirements
85	Requirements
86	Requirements
87	Requirements
88	Requirements
89	Requirements
90	Requirements
91	Requirements
92	Requirements
93	Requirements
94	Requirements
95	Requirements
96	Requirements
97	Requirements
98	Requirements
99	Requirements
100	Requirements

3.2 Agile Development Process

Description: The development process which is mainly supported by rmtoo must be agile.
Rationale: This gives a base definition for which features to add to rmtoo. Also the development process of rmtoo is also agile. (Rmtoo will be because the features are not implemented in the rmtoo).
Depends on: 2.1 rmtoo

Id	AgileDevelopment
1	AgileDevelopment
2	AgileDevelopment
3	AgileDevelopment
4	AgileDevelopment
5	AgileDevelopment
6	AgileDevelopment
7	AgileDevelopment
8	AgileDevelopment
9	AgileDevelopment
10	AgileDevelopment
11	AgileDevelopment
12	AgileDevelopment
13	AgileDevelopment
14	AgileDevelopment
15	AgileDevelopment
16	AgileDevelopment
17	AgileDevelopment
18	AgileDevelopment
19	AgileDevelopment
20	AgileDevelopment
21	AgileDevelopment
22	AgileDevelopment
23	AgileDevelopment
24	AgileDevelopment
25	AgileDevelopment
26	AgileDevelopment
27	AgileDevelopment
28	AgileDevelopment
29	AgileDevelopment
30	AgileDevelopment
31	AgileDevelopment
32	AgileDevelopment
33	AgileDevelopment
34	AgileDevelopment
35	AgileDevelopment
36	AgileDevelopment
37	AgileDevelopment
38	AgileDevelopment
39	AgileDevelopment
40	AgileDevelopment
41	AgileDevelopment
42	AgileDevelopment
43	AgileDevelopment
44	AgileDevelopment
45	AgileDevelopment
46	AgileDevelopment
47	AgileDevelopment
48	AgileDevelopment
49	AgileDevelopment
50	AgileDevelopment
51	AgileDevelopment
52	AgileDevelopment
53	AgileDevelopment
54	AgileDevelopment
55	AgileDevelopment
56	AgileDevelopment
57	AgileDevelopment
58	AgileDevelopment
59	AgileDevelopment
60	AgileDevelopment
61	AgileDevelopment
62	AgileDevelopment
63	AgileDevelopment
64	AgileDevelopment
65	AgileDevelopment
66	AgileDevelopment
67	AgileDevelopment
68	AgileDevelopment
69	AgileDevelopment
70	AgileDevelopment
71	AgileDevelopment
72	AgileDevelopment
73	AgileDevelopment
74	AgileDevelopment
75	AgileDevelopment
76	AgileDevelopment
77	AgileDevelopment
78	AgileDevelopment
79	AgileDevelopment
80	AgileDevelopment
81	AgileDevelopment
82	AgileDevelopment
83	AgileDevelopment
84	AgileDevelopment
85	AgileDevelopment
86	AgileDevelopment
87	AgileDevelopment
88	AgileDevelopment
89	AgileDevelopment
90	AgileDevelopment
91	AgileDevelopment
92	AgileDevelopment
93	AgileDevelopment
94	AgileDevelopment
95	AgileDevelopment
96	AgileDevelopment
97	AgileDevelopment
98	AgileDevelopment
99	AgileDevelopment
100	AgileDevelopment



Backlog

Priority	Requirement Id
11	8.2 Documentation of the Graph output
10	6.1 Documentation of the Graph output
8	10.1 Makefile
7	10.2 Makefile Dependencies
6	7.5 Feature - Make Auto Fill Mode
6	7.6 Feature - Make Popout Mode
6	7.3 Feature - Make Elaboration

Requirements Elaboration

Priority	Requirement Id
11	8.2 Documentation of the Graph output
7	7.7 Feature - Make to Support Traceability
7	7.1 Feature - Make
6	3.5 Easy Extensible
6	3.7 Easy Editable
5	3.3 Traceability
5	3.2 Agile Development Process
3	3.1 rmtoo must work on Requirements
3	3.1 rmtoo
3	3.1 rmtoo - Generation of Results
2	3.3 High Priority Risk
2	6.1 Output of Different Architectures
1	8.2 Documentation of the Graph output

Input Data

Command line tool

Output Artifacts

Basic Facts (1/4)

- *rmtoo* is a minimalistic non-interactive requirements management tool.
- *rmtoo* works on data stored in the file system (plain text files).
- *rmtoo* is a command line tool which reads in files and creates output.

Basic Facts (2/4)

- *rmtoo* supports different output formats and artifacts. For example:
 - PDF / LaTeX
 - XML / HTML
 - GanttProject
 - Requirements Dependency Graph
- *rmtoo* can be easily extended by adding new output modules.

Basic Facts (3/4)

- *rmtoo* data files can be handled by standard *nix commands (emacs, vi, grep, awk, streplace, sed, ...).
- *rmtoo* runs largely on the same hardware and operating system where the development is done – no need for a dedicated machine.
- *rmtoo* baselining can be handled by a version control system.

Basic Facts (4/4)

- *rmtoo* needs minimal resources
 - Netbook CPU
 - 10MByte disk space for installation and documentation
 - RAM usage dependent on size and duration of project
- *rmtoo* comes with a complete template project. Using this makes it simple to start own projects using *rmtoo*.

rmtoo does not

- *rmtoo* has no GUI.
- *rmtoo* does not come with a database.
- *rmtoo* does not have an integrated editor.
- *rmtoo* does not provide a UML editor.
- *rmtoo* does not have import capabilities (e.g. from a spreadsheet or a word processing document).

License

- *rmtoo* is Open Source.
- *rmtoo* is free.
- *rmtoo* is licensed under GPLv3.
- Commercial support is available.

4. 10 dos and 10 don'ts

reasons for using rmtoo

reasons for not using rmtoo

Use rmtoo

- when you want to be fast.
- when you want to have full control over the input files.
- when you need one set of requirements to be converted to various different output formats.
- when you need structured requirements.
- when you need a tool that supports you.

Use rmtoo

- when you need functionality which is easily extensible.
- when you want a single source of information for all requirement related materials.
- when you want the requirements for the requirements management tool you are using.

Use rmtoo

- when you want to store the requirements with the implementation source code.
- when you want to include building requirements in your continuous build environment.

Do not use rmtoo

- when a GUI is needed.
- when a web-based approach is needed.
- when you do not need full control over your data.
- when you want to adapt your behavior to a tool's processes.
- when you want to put your internal data into another companies computer.

Do not use rmtoo

- when you want to work with a tool which tries to do everything.
- when you want to be dependent on one vendor.
- when you need a traceability matrix.
- if you want to make dozens of mouse-clicks to get one output document.
- if you want to spend money.

5. Future

Future / Plans

- Some features are currently missing but planned for the upcoming releases
 - Traceability
 - Better support writing requirements in Emacs mode
- Community, User and Customer driven

Thank you!



Copyright

This document is distributed under the creative commons license 'Attribution-Noncommercial-No Derivative Works 3.0 Germany'

© 2010 flonatel GmbH & Co. KG

<http://www.flonatel.de/projekte/rmtoo>