




The Devil's List



Introduction

- Main purpose.
- Some stakeholder's lists.
- Specialists (Only they know) 


In a construction project they are the engineers and architects.
In a legal project they are the lawyers.
In an IT project they are the programmers.

- In other words: your headache.

The Devil's List





What

- Undocumented pending tasks that must be completed. 
- Usually related to what the person sense of priority is.
- Many times depends on the actual state of mind of the person.
- Sometimes they are not specifics (May be abstract).
- Always is part of a bigger task.

The Devil's List



Where

- In our mind. 
- Therefore it exist in every discipline.
- Therefore it is discipline related.
- Some people consider them as personal. 
- If the task is part of the project then:
 - It is not personal.
 - It is not private.
 - Must be shared.

The Devil's List




When

- The minute we think of a problem that requires a solution, we think of tasks to be done.
- Sometimes we hear them from other participants or from someone with experience from a similar project.
- A previous thought that was forgotten and came back. ⚡

The Devil's List




Why

- Because our mind goes faster than our hands.
- Many times we are in situation where we can not commit the task to a list. 
- Sometimes we have so many lists that it becomes impossible to manage them.
- The tasks may not have an easy solution and it must be investigated (spike).

The Devil's List



History


- User's stories (tasks) almost completed.
- Small tasks to be completed as soon as possible (same day?).
- Tasks that are 99% completed, the rest is easy. 
- These mini-tasks usually have a wrong time estimate. (low)
- **Question** Why if the tasks are small and simple they are not completed.
- **Answer** Probably because:
 - they are not a few,
 - or they are not small,
 - or they are not simple,
 - or all of the above.

.. and the rest is history.

The Devil's List



Recap

- If a pending (required) task is not documented, 
 - must probably it will be forgotten
 - or it will reappear at the worst possible moment.
- We must have a sense of urgency when dealing with the list.

.. pay the shilling to the King.

The Devil's List



- Lets talk about some examples.
- These examples are not real, but are taken from stories found in the internet about some IT projects with bad results.
- But I am sure that similar situations are found in any discipline.
- IT presents a special problems because of the magnitude of the small pieces that must be put together before a system becomes operational.
- If you try to document each and everyone of the pieces you will never begin the project.
- Because of the everyday changes in the IT industry the problem is compounded incredibly.
- In our business there is never a single way of solving a problem.

The Devil's List




Business Requirement:

- Build a Web page where clients can add their information.

Functional Specifications:

- Entered data must pass basic editing criteria.
- The email field must be used as key.



Programming Phase:

- Somebody realized that if the email is not valid, the activation code that is sent to the client will never be received.
- The programmer thinks of a solution: 
Validate the email; force the reentering by locking any paste in that field; allow email changes the first time only; etc.

The Devil's List



Reality:

- Because all of this happened during a long weekend in Playa Santa (Saturday night while looking at the moon), and because it was a piece of cake, and because he knew exactly how to do it, etc, etc, etc.
- Next working day (Tuesday) the office is in chaos, the server is down and there is big meeting (requested by the PM) asking for explanations why the project is behind schedule (rarely happens).
- Either because the programmer forgot or would not dare to raise the issue, the task is postponed to be completed another day (and tomorrow never came). 
- Usually this task is never completed and will come up during the testing phase.
- Sometimes it is never implemented, and that is why we have so many systems that do not meet the minimum standard. 

The Devil's List



Business Requirement:

- Build an Accounting Module to automatically generate G/L Trans as part of an application.

Functional Specifications:

- System must be table driven.
- Accounts and transactions type will be maintained by the user.
- Not all users that worked in the accounting section participated.



Programming Phase:

- The system is built as per specification

Testing Phase:

- The system is tested as per specification

Documentation:

- The system is documented as per specification



The Devil's List



Reality:

- Many times users do not see the system until it is completed and ready for acceptance.
- This module is part of a bigger module, so it will be tested last.
- During testing of the main application the user executes an activity that calls for voiding accounting generated the previous day.
- The accounting module was not prepared to handle this scenario.
- Every body runs to update the functional specifications to include the new scenario.
- The testing of the application must be interrupted until the required changes are completed and ready to be tested.
- Because of this crisis, there is a high probability that other features not so important will be overlooked.

The Devil's List



Consequences

- It was obvious.
- If you just knew the system.
- I said it, but nobody listened.
- It is not my responsibility.
- Somebody else works with that.
- Is not my fault, but without these changes the system is of no use.
- Nobody ask my about that.
- I explained how I do it now, but I did not know that I could improve it. ●

The Devil's List

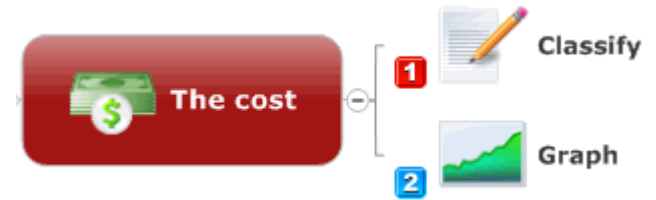


Can it be avoided?

Can it be avoided?

- No! It cannot be avoided. ⚡
- Can it be minimized? Yes. It can be minimized.
- How?: Documenting, Documenting, Documenting. ●
- Before you can solve/minimize a problem, you must recognize the problem.
- If you recognize that you as a PM will have the devil's list problems, then you can begin preparation to discover them and document them.
- In a way, you are flying blind in a situation where many of the participants are hiding (some inadvertently) the airports (tasks) from you.
- Because you do not know the specific tasks that have been identified but not documented, makes you impotent to avoid the problems that will arise. ●

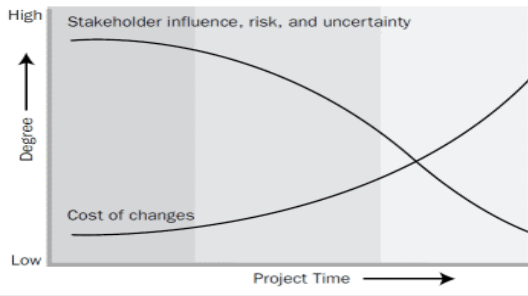
The Devil's List



Classify

- The biggest problem with the devil's list is the cost involved. (money, time, quality, prestige, good will, etc.)
- The real cost is never known. But you can be sure that it is huge.
- Lets talk about the some of the elements of the list:
 - The nice to have's (mostly in the minds of the programmers). ●
 - The tasks that add value to the product.
 - The tasks that are show stoppers. ●
- In addition we have:
 - The tasks that never surfaced.
 - The tasks that surface early in the project.
 - The tasks that surface late in the project.
 - The tasks that are detected during/after implementation.

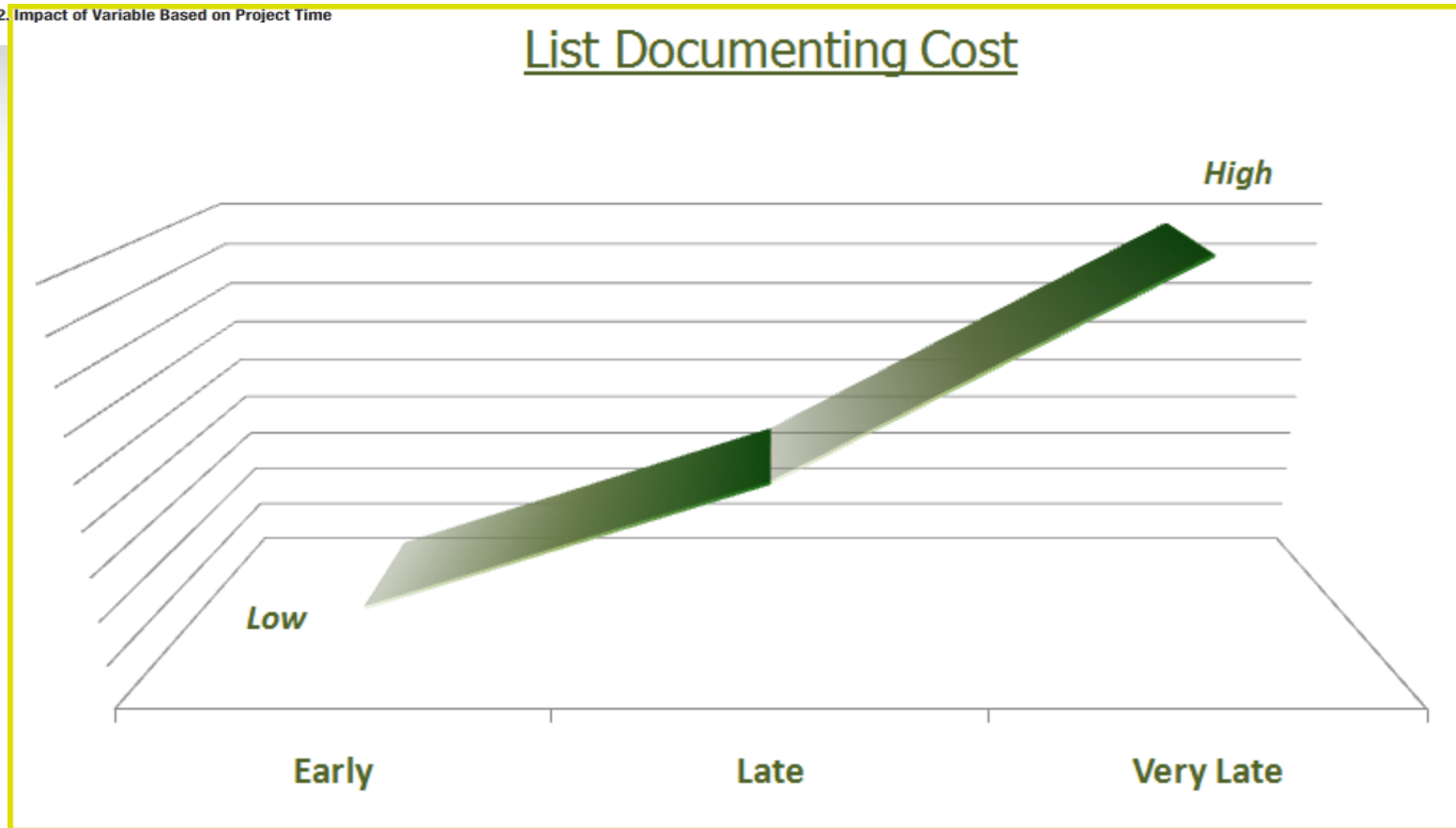
The Devil's List



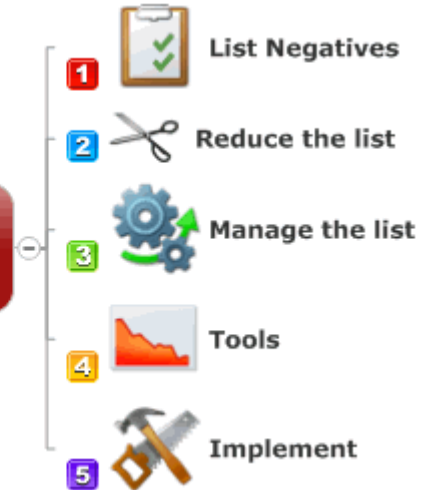
Pasted from ([PMBOK Guide](#))



Figure 2-2. Impact of Variable Based on Project Time



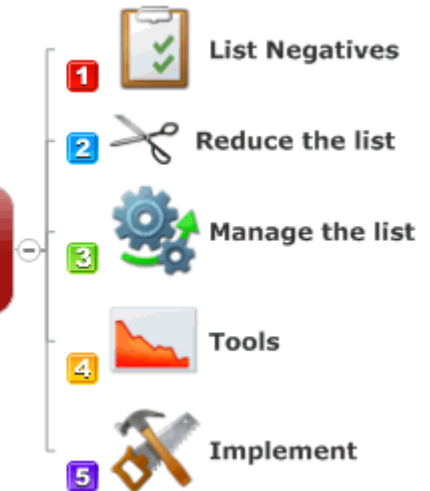
The Devil's List




Negatives

- Cost increase
- Schedule extension
- Frustration
- Product does not meet a standard
- Distrust of the product
- Friction between the users, builders and PMs
- Finger pointing

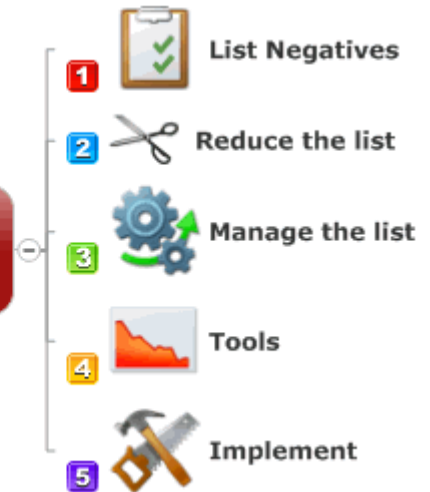
The Devil's List



Reduce it

- Recognize that this list is:
 - A very special list
 - It is powerful
 - It is always there
- Find a way to convince the project team to come forward and publish identified tasks that have not been documented.
- Be aware what "a punto de caramelo" means: 
 - A lot of unidentified tasks
- Any activity that is 97-99% completed must be examine.

The Devil's List



Manage it

- Regular meetings similar to scrums (very short)
 - What tasks has been burned since last meeting
 - Present new tasks that have been or will be included in the list
 - What tasks will be burned before next meeting
 - What impediments are outstanding
- Manage this list separately (Use a tool like KanBan)
- Take advantage of slack times
- Create cross functional teams
- Do it with passion

Agile

The Devil's List








- 1 List Negatives
- 2 Reduce the list
- 3 Manage the list
- 4 Tools
- 5 Implement

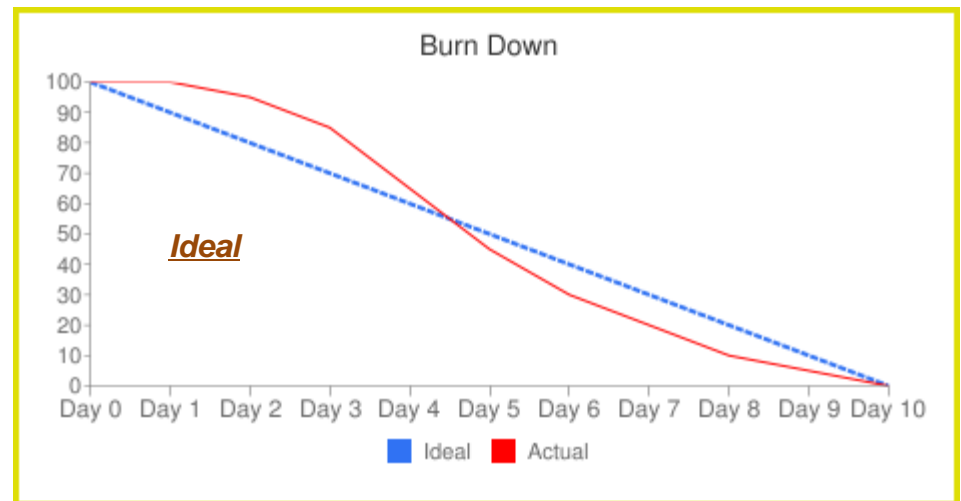
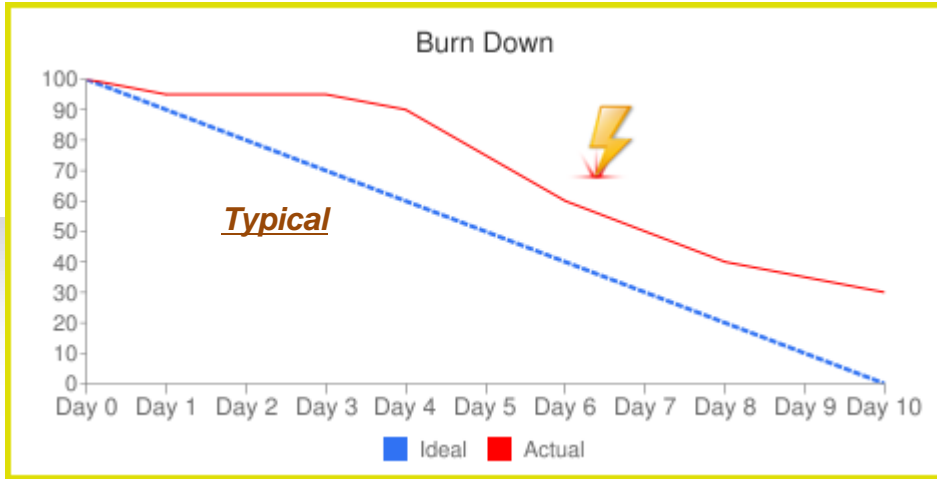
Pasted from <http://agilezen.com/>

Agile

The Devil's List

 **How to reduce its negative impact**

- 1  List Negatives
- 2  Reduce the list
- 3  Manage the list
- 4  Tools
- 5  Implement



Agile

The Devil's List



- 1 List Negatives
- 2 Reduce the list
- 3 Manage the list
- 4 Tools
- 5 Implement






Task Kanban Board

Pasted from <<http://www.infoq.com/articles/agile-kanban-boards>>

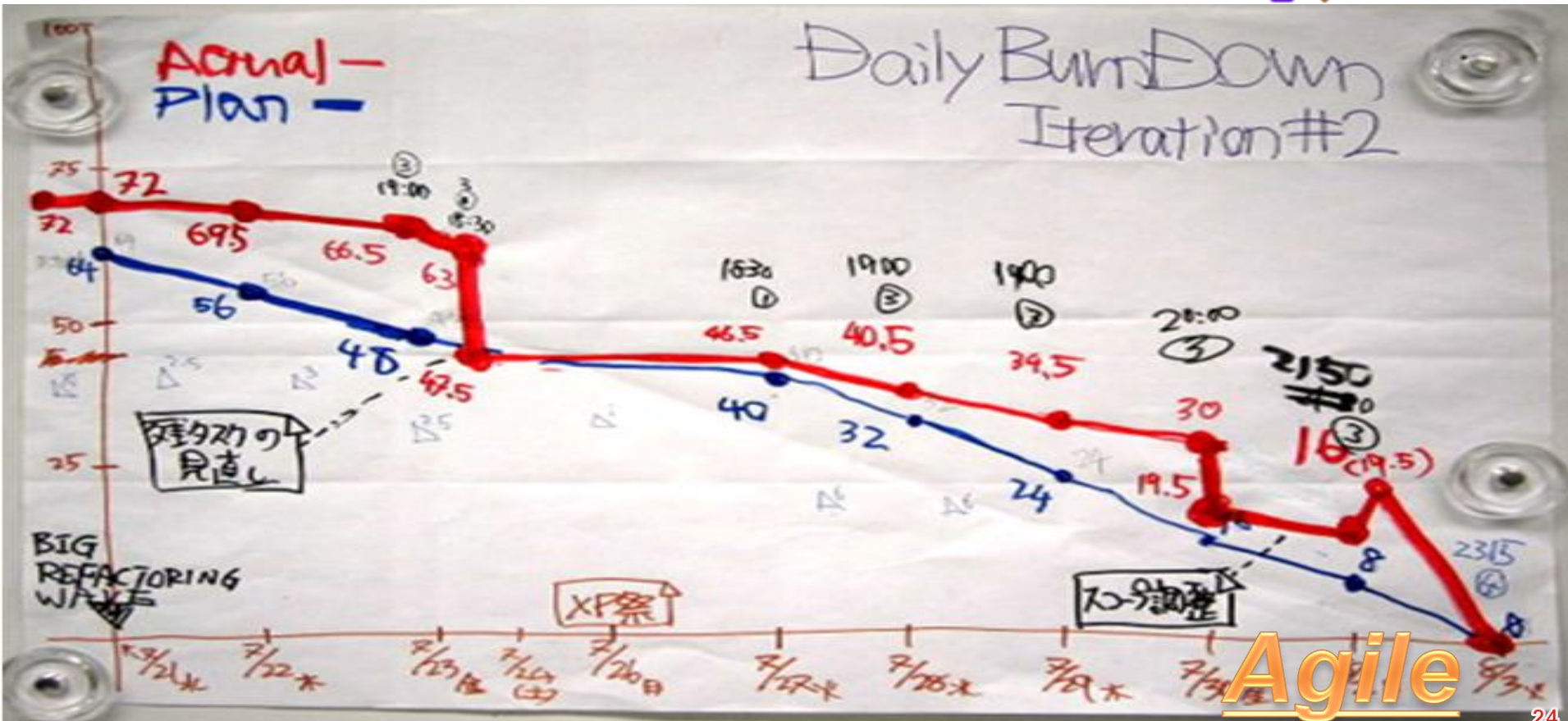


The Devil's List

 How to reduce its negative impact

- 1  List Negatives
- 2  Reduce the list
- 3  Manage the list
- 4  Tools
- 5  Implement

Pasted from <<http://www.infoq.com/articles/agile-kanban-boards>>



Agile

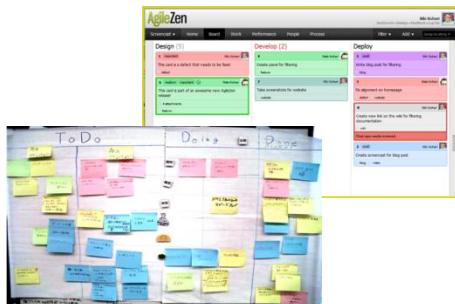
The Devil's List



- 1 List Negatives
- 2 Reduce the list
- 3 Manage the list
- 4 Tools
- 5 Implement



Document
Approve
Priority
Assign
Schedule



Agile

The Devil's List

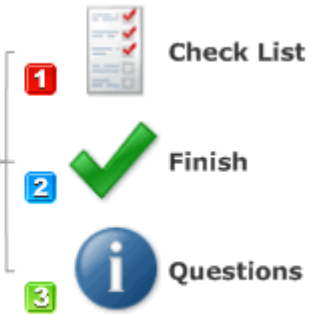


Check List

- Accept that there is a real problem (Clear and Present Danger)
- Prepare yourself to handle it
- Make “alianzas” with the participants and try to educate them about the negative impact of these lists
- Classify the tasks as nice to have, etc
- Obtain approval and set priorities
- Update/Manage the KanBan
- Regular meetings similar to scrums

Agile

The Devil's List



I wanted to present something that we have in all projects even if we do not know it.

Move all the tasks
from the **Devil's List**
to the
Angel's List.



Thank You very much.

The Devil's List



Questions

