

Tasks / Issues / Action Items

This tutorial/white paper explains the concept and ideas behind the ***pmodbase*** feature 'Tasks / Issues / Action Items', and how to employ this feature in your project.

Introduction to the Problem

The concept of an Action Item List, List of Open Points, Issue Log, Task List appears to be trivial and is often managed by a simple spreadsheet or a table in a document:

- collect various items: Actions, open points, issues, tasks, problems
- describe the item: What is the task to be done? What is the issue to be resolved? What action needs to be taken?
- assign the item to someone responsible to lead this item so it can be 'closed' or 'resolved'.

When attending various meetings, different lists (one for each type of meeting) are created –seldom by the same person. So they will look differently, might contain more or less information, and may even serve slightly different purposes.

To inform the members of the meeting about the open items, that list (the spreadsheet or table in the minute of meeting) is sent around to all participants, either for information, or for action (on the items assigned to the receiver).

I have often seen that the list is appended by additional attributes over the time. Other managers make status notes to keep track over the evolution of the issue, and also to remember the previous status.

It takes a very tidy person to keep track of all these action items in those different lists: You have to remember all items assigned to you, remember the status for the next meeting (when the list is retrieved and status reports are requested), and still have your mailbox cluttered with copies of the lists.

Concept of the ***pmodbase*** Solution

What solution does ***pmodbase*** offer for handling such lists in a multi-user, multi-group environment? How can ***pmodbase*** help to increase the efficiency of the project management here?

Definitions

pmodbase does not distinguish between Action Item Lists, Issue Logs, Task Lists, List of Open Points. Such lists serve the same purpose: To keep track of items of interest over time. Therefore, this feature was named **Task / Issue / Action Items**. A list of such items is then named **Task / Issue / Action Item List**.

The Lists

First of all, I assume that projects (or people working in teams) have more than a single list of such items: There is at least one list for each group having meetings (as these lists are a useful tool to manage meetings efficiently). **pmodbase** will provide you and your team with any number of such lists: Use one list for project A, another for project B, one for group internal affairs, one for your own needs, one for issues, one for action items, etc.

Each list will have an owner, that is the person who normally manages that list and the items in that list, often the manager leading the meeting.

Certainly, that manager has a deputy who can also manage that list on her¹ behalf, either during the meeting or when she is out of office.

Lists have a descriptive label and a short, unique code to be used as prefix for the item numbers such that items in one list can easily be distinguished from other items just by looking at this prefix. For example, action item 'ABC-xyz' is item 'xyz' in the list using the prefix 'ABC'.

The Items

Each list consists of items: tasks, action items, issues. Each item should be described as clearly as possible such that every participant of the meeting understands what is expected to solve / close this item. Then a member of the group would be assigned to take care that this item is resolved. This member is called 'owner' of that item.

Each action item has a unique number within the respective list: This number is prefixed by the code specified for that list.

Other attributes for tasks, action items, issues can be set: a priority (high, medium, low) and an expected – or better: a mutually agreed – completion date ('due date').

Following the 'assigned owner and responsibility' principle of **pmodbase**, only the owner of the list or his deputy is permitted to modify the attributes of any item in that list, no one else. This enforces clear communication channels: Any change has to go through the person responsible for all items in that list.

The Members

The responsible owner of an item can only be selected from the members of the meeting group: This is to ensure that no person outside of the group can be assigned ownership/responsibility for a specific item. This may sound somewhat restrictive but it is proper practice to have someone in the (meeting's) group control the progress and report the final conclusion or resolution of the item, as he or she is aware of the situation in which the item was discussed and logged.

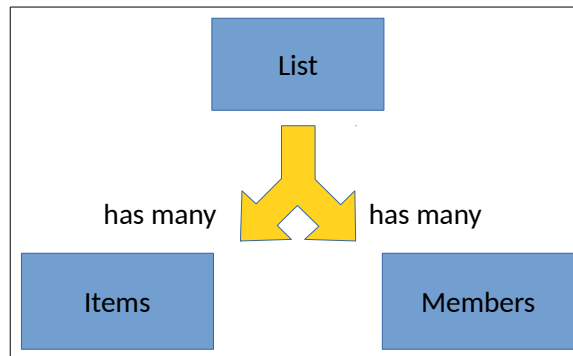
1) As I found it difficult to write comprehensive but gender-neutral text, so I decided to use the female form for the manager and the male form for members of her group: However, this shall not exclude the other gender for any of these positions.

Members of the team or the group invited for the meeting are selected from the project's participants.

Members of that group may be given specific access rights to the items of that list: a member may be restricted to view any items, or he may be allowed to comment on items assigned to him.

The Data Structure

These considerations led to the following structure of this feature in ***pmodbase***:



Each Task / Issue / Action Item List consists of the various Task / Issue / Action Items.

There is a number of people who are allowed to access (read) that list and the associated items, these are the Task / Issue / Action Item List Members.

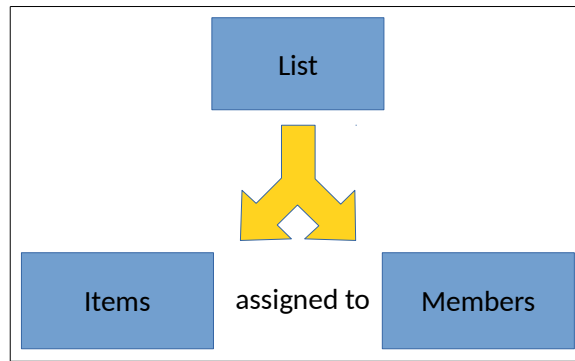
Any user with the respective rights can create any number of Task / Issue / Action Item Lists. The creator of the list is automatically the owner of that list.

The owner of the list can assign a deputy (user) to this list who has the same rights as the manager (in respect to that list). The owner may also transfer the ownership of the list to another user – not necessarily her deputy.

The manager (or the deputy) then assigns other users as member of the list. These members are usually the members of the meeting group for which this list was created.

Workflow

Tasks / Action Items / Issues are documented in the meeting and the ownership of an item is assigned to a member of that group. For action items, the owner of an action item would be responsible to ensure that the action is followed up and finally closed. For issues, that person would be responsible to lead the resolution of that issue.



The ownership of an item by an individual member of the group shall ensure that there is single, real person responsible for closing this item. Even if more than one person may be required to solve the issue, a single person should be accountable for driving the action and report the status in future meetings.

Again the 'assigned owner and responsibility' principle is used: An owner of an item is responsible for that item until the item is resolved, or the owner of the list (the manager) reassign the item to someone else.

The owner of an item is only permitted to comment on the (status of the) item but he is not allowed to modify the task or change any other attribute of the item (description, priority, due date).

In practice this will lead to a strict process of handling tasks, action items, and issues in a group:

- An item is created during a meeting
- The item is processed by the responsible owner, possibly a comment is added to the item regarding the status or progress reached.
- During the next meeting, all open items are reviewed: If an item has progressed sufficiently to be closed, this is then acknowledged by the manager (of the list) by setting the status of the item to 'closed'.
- If any attributes of the item, for example the description of the task, action, issue, needs to be adjusted, the due date to be extended, or the responsibility to be reassigned to another person, this can only be done in cooperation and with the knowledge by the owner of the list – as he is the only one who has permission to do so.
- If an item requires management attention, this could be done 'offline' (outside the meeting) and the results can be added to the list.

History

Self-made Task / Issue / Action Item lists are often extended to hold the history of events: When was this item opened? Who was responsible over the time? How did the description of the issue change?

pmodbase includes a history feature for these items: All changes are tracked and stored in the database for review. The changes are listed in reverse chronological order and include only the modified items.

Information Privacy and Co-Determination

The Tasks / Issue / Action Item Lists were designed to support (project) management tasks on a state-of-the-art, collaborative, multi-user platform. This feature was not designed to monitor project/group members and to evaluate their individual progress. Technically, it would be possible to analyse the duration of items and the amount of time a specific person was responsible for the resolution of the item. However, this never was and is not the intention of this tool. Person-related statistics are not collected.

Closed Items

Once an item is 'closed', it may need to remain visible for some time and as reference for information.

A 'closed' item can be marked 'archived' to exclude this item from future listings. However, it will remain the database but will not be visible any more.

Keywords

Action Items, Issue Log, Task List, List of Open Points, Collaboration Software, Management Tool