



# User Help

Fabasoft Scrum

Copyright © Fabasoft R&D GmbH, Linz, Austria, 2021.

All rights reserved. All hardware and software names used are registered trade names and/or registered trademarks of the respective manufacturers.

No rights to our software or our professional services, or results of our professional services, or other protected rights can be based on the handing over and presentation of these documents.

# Contents

<b>1 Introduction</b>	<b>5</b>
<b>2 Getting Started With Fabasoft Scrum</b>	<b>5</b>
<b>3 Structuring Elements</b>	<b>6</b>
<b>4 Roles</b>	<b>6</b>
<b>5 Scrum (Personal Dashboard)</b>	<b>7</b>
5.1 Dashboard	7
5.2 Actions	8
<b>6 Scrum Artifacts</b>	<b>8</b>
6.1 Story/Defect/Impediment	9
6.1.1 Dashboard	9
6.1.2 Properties	9
6.1.3 Actions and Context Menu Commands	14
6.2 Task	16
6.2.1 Properties	16
6.2.2 Actions and Context Menu Commands	17
6.3 Sprint	17
6.3.1 Dashboard	17
6.3.2 Properties	18
6.3.3 Actions and Context Menu Commands	20
6.4 Product Version	20
6.4.1 Properties	20
6.4.2 Actions and Context Menu Commands	21
6.5 Release	21
6.5.1 Dashboard	21
6.5.2 Properties	21
6.5.3 Actions and Context Menu Commands	22
<b>7 Scrum Center</b>	<b>22</b>
7.1 Dashboard	22
7.2 Settings	23
7.2.1 "General Settings" Tab	23
7.2.2 "Scrum Center" Tab	24
7.2.3 "Feature Administration" Tab	25

7.2.4 "Predefined Values" Tab .....	25
7.2.5 "Structuring" Tab .....	25
7.2.6 "Tagging" Tab.....	26
7.2.7 "User Interface" Tab .....	26
7.2.8 „Logos“ Tab .....	26
7.2.9 "Default Values" tab .....	26
7.3 Actions .....	27
<b>8 Scrum Project</b> .....	<b>27</b>
8.1 Dashboard .....	27
8.2 Settings .....	28
8.2.1 "General Settings" Tab .....	28
8.2.2 "Scrum Project" Tab.....	29
8.2.3 "Initialization" Tab.....	30
8.2.4 "Predefined Values" Tab .....	30
8.2.5 "Structuring" Tab .....	31
8.2.6 "Tagging" Tab.....	31
8.2.7 "Default Values" tab .....	31
8.3 Actions .....	32
<b>9 Additional Tasks</b> .....	<b>32</b>
<b>10 Administration Tasks</b> .....	<b>33</b>
10.1 Prerequisites .....	34
10.2 Scrum (Personal Dashboard).....	34
10.2.1 Scrum Center .....	34
10.2.2 Scrum Project.....	35
<b>11 Web Service</b> .....	<b>35</b>
11.1 SOAP and WSDL.....	35
11.2 JSON .....	36
11.3 Available Operations .....	36

## 1 Introduction

Scrum is an iterative, incremental, agile software development model focusing on business needs. Fabasoft Scrum helps scrum teams to self-organize and work together.

If you are not familiar with scrum, have a look at the basic concepts before reading further (e.g. [https://en.wikipedia.org/wiki/Scrum\\_%28software\\_development%29](https://en.wikipedia.org/wiki/Scrum_%28software_development%29)).

## 2 Getting Started With Fabasoft Scrum

### Administration

To be able to set up your own scrum projects the “Scrum” app has to be assigned to your organization. If you are an organization owner, organization administrator, payment user or scrum administrator, you will find topics on setting up and administering scrum projects in chapter 10 “Administration Tasks”.

### Usage

It is assumed that a scrum center and a scrum project are available and the scrum project team is already defined. The starting point is the personal scrum dashboard.

- Team members and stakeholders create stories (requests for new features) and defects.
- Team members may create impediments, if there are circumstances that prevent them from working as efficiently as possible.
- New stories are collected in the *Inbox*, defects in the *Defects* and impediments in the *Impediments* widget.
- All team members may accept (“Accept” action) stories, defects and impediments, but only the product owner can reject these backlog items. Accepted stories are moved from the *Inbox* to the *Backlog* of the corresponding scrum project.
- The product owner prioritizes the stories (*Order* field).
- The scrum project is filled with requirements, now the work should start. To do so, a team member creates a sprint with a defined start and end date.
- Stories that are in the backlog of a scrum project can be added to a sprint (“Add to Sprint” context menu command). Thereby the effort of a story has to be estimated in story points by the team. This helps to decide how many stories can be finished within the sprint.
- Defects and impediments can be added to the sprint by team members as well.
- To structure the work on a story or defect, tasks can be defined.
- After adding the desired stories to the sprint the scrum master or product owner starts the sprint (“Start Sprint” action).
- The team members work on the stories and defects. Depending on the progress of the tasks and stories the corresponding state (e.g. to verify) is set by the team members. At the end of the sprint hopefully all stories have the state “Done”.
- The product owner reviews the working results of the completed stories and releases or not releases them.
- The scrum master or product owner ends the sprint (“End Sprint” action) and a new sprint can be started.

### 3 Structuring Elements

Fabasoft Scrum provides the following structuring elements:

- **Scrum Center**  
The scrum center allows defining general settings that apply to all subordinate scrum projects. The team members of a scrum center either have special access rights on the scrum projects or can be assigned to the scrum projects as team members.  
**Note:** On the basis of the defined users the consumed licenses are calculated.
- **Scrum Project**  
Each scrum project is assigned to a scrum center. The scrum project defines the scrum team and provides a central repository for all scrum artifacts.
- **Sprint**  
A sprint is the basic unit of development in scrum. In general a sprint lasts a few weeks.
- **Story**  
A story is used to describe requirements from a user's perspective.
- **Defect**  
A defect is used to describe an error in the product.
- **Impediment**  
Anything that prevents a team member from performing work as efficiently as possible.
- **Product Version/Release**  
Stories and defects can be assigned to specific product versions and releases.

### 4 Roles

Fabasoft Scrum provides the following roles:

#### Roles in the scrum center

- **Scrum Administrator**  
A scrum administrator can administer the scrum center and all assigned scrum projects. Typically, the scrum administrator defines the members of the scrum center. These members can be defined as scrum team in the assigned scrum projects. Additionally, a scrum administrator can define the security settings of the scrum center and all assigned scrum projects and can create new scrum centers.
- **Privileged User**  
A privileged user can administer the scrum center and all assigned scrum projects, but is not allowed to define the team members or security settings of the scrum center and assigned scrum projects.
- **Scrum User**  
A scrum user has read access on the scrum center and all assigned scrum projects that are not restricted. Scrum users can be added to the team of scrum projects they should work on.
- **Stakeholder**  
A stakeholder can create stories and defects in not restricted scrum projects and has read access on these stories and defects. Additionally, stakeholders can be assigned to arbitrary backlog items to allow them tracking these items.

#### Roles in the scrum project

- **Product Owner**  
A product owner can administer the scrum project, can define team members and has special rights like releasing and rejecting stories. Only users that are defined in the scrum center can be added as team members. To allow working with several partners within a single scrum center, the scrum center users are not suggested for selection (the whole e-mail address has to be entered).
- **Scrum Master**  
A scrum master can administer the scrum project and can define team members. Only users that are defined in the scrum center can be added as team members. To allow working with several partners within a single scrum center, the scrum center users are not suggested for selection (the whole e-mail address has to be entered).
- **Team Member**  
A team member can work within the scrum project.
- **Stakeholder**  
The stakeholder role is only available in restricted scrum projects, because stakeholders defined in the scrum center are not allowed to access restricted scrum projects.

## 5 Scrum (Personal Dashboard)

The personal dashboard is the central access point to scrum. When a user is added to a scrum center, a dashboard is automatically created and put on the user's "Home". If a user is no longer member of any scrum center, the dashboard will be removed.

### 5.1 Dashboard

The personal dashboard is divided into following areas:

- *Task Burndown*  
Shows the percentage of open tasks from all sprints in progress (only available if there are sprints in progress that contain tasks).
- *Defects*  
Shows all open defects.
- *Current Stories and Defects*  
Shows all stories of scrum projects the user is team member of and that are assigned to a running sprint. In addition, it shows all defects of scrum projects the user is team member of and that are not completed.
- *My Tasks*  
Shows all tasks in running sprints assigned to the current user.
- *Inbox*  
Shows all new stories of scrum projects the user is team member of and that are not yet accepted or rejected. Accepted stories are moved to the backlog of the scrum project. Rejected stories are moved to the backlog done of the scrum project.
- *Tracklist*  
The tracklist is a personal space where users can store backlog items they are interested in. Backlog items that are created by the user are automatically stored in the tracklist. To manage the tracklist the actions "Add to Tracklist" and "Remove From Tracklist" are provided for backlog items.

- *Product Backlog*  
Shows all accepted stories of scrum projects the user is team member of and that are not completed.
- *Impediments*  
Shows all impediments of scrum projects the user is team member of and that are not completed.
- *My Sprints*  
Shows all sprints of scrum projects the user is team member of and that are in planning or in progress.
- *Scrum Projects*  
Shows all scrum projects the user has access to. If no state symbol is shown the user has read access, if a group symbol is shown the user is a team member and if an information symbol is shown the user has added the scrum project to the statistics (personal dashboard > "Settings" action > *Show Statistics for*).
- *Scrum Centers*  
Shows all scrum centers the user has access to.

## 5.2 Actions

Depending on your roles you can execute following actions:

- **Create Story**  
Creates a new story.
- **Create Defect**  
Creates a new defect.
- **Create Impediment**  
Creates a new impediment.
- **Create Sprint**  
Creates a new sprint.
- **Create Scrum Project**  
Creates a new scrum project (only available for administrators). For more information see chapter 10.2 "Scrum (Personal Dashboard)".
- **Create Scrum Center**  
Creates a new scrum center (only available for administrators). For more information see chapter 10.2 "Scrum (Personal Dashboard)".
- **Settings**  
Opens the settings of the dashboard. Here scrum centers and scrum projects can be included or excluded from the personal dashboard. In addition, it can be defined whether newly created backlog items are added automatically to the tracklist. On the "Logos" tab you can define your own logo for the dashboard.

## 6 Scrum Artifacts

In the following chapters the basic scrum artifacts are described in detail.

## 6.1 Story/Defect/Impediment

Stories, defects and impediments are the items the scrum team works on.

### 6.1.1 Dashboard

The dashboard is divided into following areas:

- *Story/Defect/Impediment*  
Shows the content of the story, defect or impediment.
- *Acceptance Criteria* (only for stories)  
Shows the acceptance criteria text of the story.
- *Tasks*  
Shows all tasks that belong to the story, defect or impediment.
- *Documents*  
Shows all documents that belong to the story, defect or impediment.
- *Subordinate Items*  
Subordinate items of a story, defect or impediment can be belonging stories, defects or impediments. This widget is only shown when subordinate items are defined ("Relations" tab > *Subordinate Items*).
- *Affected Stories/Defects* (only for impediments)  
Shows which stories or defects are affected by the impediment. This widget is only shown when affected items exist ("Relations" tab > *Affected Items*).

### 6.1.2 Properties

A backlog item provides following properties.

#### 6.1.2.1 "Story"/"Defect"/"Impediment" Tab

- *ID*  
The identifier of the backlog item is a combination of the short form of the scrum center or scrum project and a running number. If the short form is defined in the scrum center, the ID is assigned automatically when the backlog item is added to the inbox. If the short form is defined in the scrum project, the ID is assigned automatically when the backlog item is accepted.
- *Requested by*  
Defines a user who requests the backlog item (by default the creator of the backlog item).
- *Priority*  
One of four different priorities can be selected for a backlog item (from "must have" to "nice to have").
- *Context*  
Defines an organization, organizational unit or support request the backlog item is related to.
- *Feature* (only for stories)  
Defines a story that represents the superordinate feature of the current story. If a *Feature Project* is defined in the scrum project, only stories from the feature project can be selected.
- *Topic*  
If backlog item topics are defined in the scrum center or scrum project ("Structuring" tab > *Topics for Structuring Backlog Items*), a topic can be assigned to the backlog item.

- *State*  
Shows the current state (“Received”, “Planning”, “In Progress”, “To Verify”, “Done”, “Released”, “Not Released”) of the backlog item.
- *Size*  
Stories have to be estimated when adding them to a sprint. By default story points are not used for defects and impediments but they can be enabled in the scrum center (“Scrum Center” tab > *Allow Story Points for Defects and Impediments*).
- *Size (Accumulated)*  
If a backlog item is defined as feature of other backlog items the sum of story points of these backlog items is shown in this field.
- *Subject*  
The name of the backlog item.
- *Story/Defect/Impediment*  
Contains the description of the backlog item.
- *Acceptance Criteria* (only for stories)  
Contains the conditions that must be met such that the backlog item can be released.
- *Remarks*  
Shows all remarks of the backlog item.
- *Occurred With* (only for defects)  
Defines a product version, release and build number in which the defect was found.
- *Requested for*  
The backlog item can be requested for a concrete product version and release.
- *Completed With*  
Defines the product version, release and build the backlog item was completed with.
- *Change Log*  
A free-text field that can be used to describe the changes the backlog item causes.

#### 6.1.2.2 “Details” Tab

- *Scrum Project*  
Shows the scrum project the backlog item is assigned to.
- *Sprint*  
Shows the sprint the backlog item is assigned to.
- *Product Version*  
Defines the product version for which the backlog item should be completed. Product versions can be defined in the scrum center or scrum project.
- *Release*  
Defines the release for which the backlog item should be completed. Releases can be defined in product versions.
- *Lead*  
Defines a team member that should take the lead.
- *Feature Team*  
If feature teams are defined in the scrum project (“Scrum Project” tab > *Feature Teams*), a feature team can be assigned to the backlog item.

- *Order*  
Defines the priority of the backlog item relative to other backlog items. This value can be set manually or automatically when fixing the backlog order ("Fix Order" context menu command).
- *Business Value* (only for stories)  
Defines the business value that can be generated by implementing the story.
- *Allocated on/at*  
Shows the date and time the backlog item was allocated to the scrum project. The allocation can be changed with the "Grooming" > "Change Allocation" context menu command.
- *Processing Time Since Allocation*  
The time since the last allocation.
- *Total Processing Time*  
The time since the backlog item was created.
- *Last State Change on/at*  
The date and time when the state was lastly changed.
- *Current Stage*  
If processing stages are defined for tasks, the current stage is displayed. Processing stages can be defined and ordered correspondingly in the scrum center or scrum project ("Predefined Values" tab > *Processing Stages*). Once all tasks of a particular processing stage are "Done", the story progresses to the next stage.
- *Category* (only for stories)  
For stories one of the following categories can be defined: "Story" (default), "Spike", "Constraint" and "Risk".
- *Severity* (only for defects)  
If defect severities are defined in the scrum center or scrum project ("Predefined Values" tab > *Defect Severities*), a severity can be assigned to the defect.
- *Hierarchy Level*  
If hierarchy levels (e.g. software layers) are defined in the scrum center or scrum project ("Predefined Values" tab > *Hierarchy Levels*), a hierarchy level can be assigned to the backlog item.
- *Impact Areas*  
If impact areas are defined in the scrum center or scrum project ("Predefined Values" tab > *Impact Areas*), an impact area can be assigned to the backlog item.
- *Affected Applications*  
If affected applications are defined in the scrum center or scrum project ("Predefined Values" tab > *Affected Applications*), an affected application can be assigned to the backlog item.
- *Administrative Category*  
If administrative categories are defined in the scrum center or scrum project ("Predefined Values" tab > *Administrative Categories*), an administrative category can be assigned to the backlog item.
- *Stakeholders*  
If stakeholders are defined in the scrum center team or scrum project team, a stakeholder can be assigned to the backlog item. This way stakeholders have read access on the item and can follow the working progress.
- *Terms*  
Defines keywords for the backlog item.

- *References*  
Defines documents that are relevant for the backlog item.
- *Import ID*  
An alphanumeric text to be use as a reference to an external application.

### 6.1.2.3 “Retrospective” Tab (Defect)

- *Introduced With*  
Defines a product version, release and build number in which the defect was introduced.
- *Size for Fixing*  
Defines the number of story points needed to fix the defect.
- *Story From Team’s Standpoint*  
Defines whether it is rather a story than a defect from the team’s point of view.
- *Reasons*  
If reasons are defined in the scrum center or scrum project (“Structuring” tab > *Topics for Structuring Reasons of Defects*), reasons can be assigned to the defect.
- *Causes*  
If causes are defined in the scrum center or scrum project (“Structuring” tab > *Topics for Structuring Causes of Defects*), causes can be assigned to the defect.
- *Description of Cause*  
A free-text field to define the cause of the defect.
- *Description of Measures to Avoid Similar Defects*  
A free-text field to define the measures to avoid similar defects in the future.
- *Causing Scrum Project*  
Defines a scrum project in which the defect was introduced.
- *Predictable*  
Defines whether the defect could be found by the quality assurance.
- *Causing Stories*  
Defines a list of stories that caused the defect.
- *Original Requirements*  
Defines a list of requiring stories that lead to the defect.

### 6.1.2.4 “Retrospective” Tab (Impediment)

- *Resolution*  
A free-text field to define the resolution of the impediment.

### 6.1.2.5 “History” Tab

The history tab shows events that occurred in the backlog item.

### 6.1.2.6 “Repository” Tab

On this tab the source code repository check-ins can be tracked.

### 6.1.2.7 “Repository Information” Tab

On this tab the check-ins tracked on the repository tab are parsed and displayed as list. Necessary format of the check-in messages:

```
Revision: <number>
Author: <author>
Date: <date>
Message:
<message including merged revisions (e.g. r12345)>
---
Modified: /<branch>/<arbitrary path>
```

### 6.1.2.8 “Tasks” Tab

The fields concerning the effort are only available, if the capacity planning is enabled in the scrum center (“Feature Administration” tab > *Enable Capacity Planning*).

- *Planned Effort*  
Shows the total hours planned for the backlog item based on the planned hours of all tasks.
- *Estimated Remaining Time*  
Shows the estimated remaining time based on all tasks.
- *Time Worked in Sprint*  
Shows the total time worked in the current sprint on the backlog item based on the tasks.
- *Time Worked*  
Shows the total time worked, if it has been worked on the backlog item for several sprints.
- *Planned Effort (Including Subordinate Items)*  
Shows the total hours planned for the backlog item including all subordinate backlog items.
- *Estimated Remaining Time (Including Subordinate Items)*  
Shows the estimated remaining time for the backlog item including all subordinate backlog items.
- *Time Worked in Sprint (Including Subordinate Items)*  
Shows the total time worked in the sprint on the backlog item including all subordinate backlog items.
- *Time Worked (Including Subordinate Items)*  
Shows the total time worked on the backlog item including all subordinate backlog items.
- *Tasks*  
Shows all tasks of the backlog item.

### 6.1.2.9 “Relations” Tab

The most fields on this tab are only visible, if they contain a value.

- *Feature*  
Defines a story that represents an entire feature the current story belongs to.
- *Subordinate Items*  
Defines all subordinate backlog items of a story that represents a feature.
- *Joined With*  
Shows the backlog item, the current backlog item is joined with.
- *Split From*  
Shows the original backlog item that has been split.

- *Joined From*  
Shows the backlog items, the current backlog item is joined from.
- *Split Into*  
Shows the resulting backlog item from a split.
- *Dependency for*  
Defines backlog items that are dependent on the current backlog item. This field is visible, if a backlog item defines dependent backlog items ("Relations" tab > *Dependent on*).
- *Dependent on*  
Defines prerequisites for the backlog item. A warning is displayed, if a team member starts working on a backlog item that has not completed prerequisites.
- *Provides Requirements for*  
Defines requirements for the backlog item. This field is visible, if a defect defines original requirements ("Retrospective" tab > *Original Requirements*).
- *Caused Defects*  
Defines defects that were introduced by the backlog item. This field is visible, if a defect defines the causing backlog item ("Retrospective" tab > *Causing Stories*).
- *Impediments* (only for stories and defects)  
Defines impediments belonging to the backlog item.
- *Affected Items* (only for impediments)  
Defines backlog items that are affected by the impediment.

#### 6.1.2.10 "Effort/Costs" Tab

This tab is only available if the capacity planning is disabled. When setting the state of a story to "Done" the effort can be entered.

- *Effort in Current Sprint (MD)*  
Defines how many man days it was worked on the backlog item in the current sprint.
- *Total Effort (MD)*  
Defines how many man days it was worked on the backlog item in all sprints.
- *Effort*  
A list showing all sprints with the assigned man days.

#### 6.1.2.11 "Ratings" Tab

This tab shows the rating of the backlog item. Ratings (context menu command "Rate") can be used to help prioritizing the backlog item.

### 6.1.3 Actions and Context Menu Commands

- **Accept**  
New stories (inbox), defects and impediments have to be accepted by a team member, or rejected by the product owner. Accepted stories are moved to the backlog of the corresponding scrum project. Rejected stories are moved to the backlog done.
- **Add to Sprint**  
Adds a backlog item to a sprint that is in planning or in progress.
- **Remove From Sprint**  
Removes a backlog item from a sprint.

- Add Task  
Adds a task to a backlog item.
- Add Remark  
Adds a remark to a backlog item.
- Add to Tracklist  
Adds a backlog item to the personal tracklist, to easily keep an eye on it.
- Remove From Tracklist  
Removes a backlog item from the tracklist.
- Mark as Read/Mark as Unread  
Stories in the inbox can be marked as read or unread to ease the management of them.
- In Progress/To Verify/Done/Released/Not Released  
Sets the state of the backlog item. The available state transitions depend on the current state of the backlog item and on the states of subordinate tasks. For example, a story can only be "To Verify", if all tasks have at least the state "To Verify".  
Only product owners can release or not release backlog items.
- Revise  
Revises a released backlog item that should not have been released or rejected.
- Create/Assign Impediment  
Creates a new impediment.
- Grooming
  - Change Allocation  
Assigns the backlog item to another scrum project.
  - Released  
Releases a backlog item.
  - Reject  
Rejects a backlog item that should not be implemented. Rejected backlog items are moved to the backlog done of the scrum project.
  - Merge Story/Defect/Impediment  
Merges backlog items that have a similar or equal content.
  - Split  
Splits a backlog item in two parts. Useful, for example, if a backlog item is too big such that it cannot be implemented in one sprint.
  - Change to Story/Defect/impediment  
Changes the type of a backlog item.
- Fix Order  
Assigns a number (*Order* field) to the selected items. The order in which the items are selected will be considered by the numbering. The numbering starts with 10, if for the first item no order or an order lower than 10 has been defined. The following items are numbered in steps of ten (20, 30, 40 and so on). If the first item has a defined order equal or higher than 10, the starting number will be based on this number (e.g. 34 will be rounded to 40 and the next numbers are 50, 60 and so on).
- Rate  
Rates the story (one to five stars).
- Request Hotfix  
Requests a hotfix for a defect for a certain product version and release.

- **Lock/Unlock**  
Locks or unlocks a backlog item. The content of a locked backlog item cannot be changed. Only product owners can lock or unlock backlog items.

## 6.2 Task

In general, a story or defect can be subdivided into several tasks. The state of a task reflects the working progress.

### 6.2.1 Properties

A task provides following properties.

#### 6.2.1.1 "Task" Tab

- *Backlog Item*  
Shows the backlog item the task belongs to.
- *Scrum Project*  
Shows the scrum project the task belongs to.
- *Current Processor*  
Defines the team member who works on the task.
- *State*  
Shows the current state of the task.
- *Name*  
The name of the task.
- *Description*  
A detailed description of the task might be helpful.
- *Remarks*  
Shows all remarks of the task.

#### 6.2.1.2 "Details" Tab

- *Current Processor*  
Defines the team member who works on the task.
- *State*  
Shows the current state of the task.
- *Task Topic*  
If task topics are defined in the scrum center or scrum project ("Structuring" tab > *Topics for Structuring Tasks*), a topic can be assigned to the task. The topics for structuring tasks can also be used to describe processing stages ("Predefined Values" tab > *Processing Stages*).
- *Planned Effort*  
Shows the planned effort of the task that can be defined when creating a new task. This and the next properties are only available, if the capacity planning is enabled in the scrum center ("Feature Administration" tab > *Enable Capacity Planning*).
- *Estimated Remaining Time*  
Shows the estimated remaining time of the task that can be defined when changing the state of the task.

- *Time Worked*  
Shows the total time worked on the task.
- *Time Worked in Sprint*  
Shows the total time worked on the task in the current sprint.
- *Time in Current State*  
Shows how many hours the task has been in the current state (only shown if the state lasts more than one hour).
- *Time Registration*  
Shows the history of time definitions for the task.
- *Terms*  
Defines keywords for the task.
- *References*  
Defines documents that are relevant for the task.

### 6.2.1.3 "History" Tab

The history tab shows events that occurred in the task.

## 6.2.2 Actions and Context Menu Commands

- Add Remark  
Adds a remark to the task.
- Take Task  
To avoid that several team members work on the same task, a team member can take a task.
- Planning/In Progress/To Verify/Done/Not Relevant  
Changes the state of the task. Which actions are available depends of the current state of the task.
- Copy  
Copies the task to the clipboard.
- Properties  
Opens the properties of the task.

## 6.3 Sprint

A sprint is the basic unit of development in scrum. In general, a sprint lasts a few weeks.

### 6.3.1 Dashboard

The dashboard is divided into following areas:

- *Story Burndown*  
The burndown is based on the story points of stories with state "Done". Only available if at least one backlog item is in the sprint and if correspondingly configured in the scrum project ("Scrum Project" tab > *Show Story Burndown*).
- *Task Burndown*  
The burndown is based on the amount of tasks with state "Done". Only available if at least one task is in the sprint and if correspondingly configured in the scrum project ("Scrum Project" tab > *Show Task Burndown*).

- *Hours Burndown*  
The burndown is based on planned time left and estimated remaining time. The time is evaluated considering the values in the tasks (*Planned Effort* and *Estimated Remaining Time*). Only available if the capacity planning is enabled in the scrum center ("Feature Administration" tab > *Enable Capacity Planning*) and if correspondingly configured in the scrum project ("Scrum Project" tab > *Show Hours Burndown*).
- *Task Board*  
Shows the tasks of all backlog items of the sprint.
- *Sprint Backlog*  
Shows all backlog items of the sprint.
- *Documents*  
Documents that belong to the sprint can be stored here. Folders can be used to structure the documents. By default a special folder is available that contains all documents that are assigned to the backlog items of the sprint.

## 6.3.2 Properties

A sprint provides following properties.

### 6.3.2.1 "Sprint" Tab

- *Name*  
The description of the sprint.
- *Scrum Project*  
Shows the scrum project the sprint is assigned to.
- *State*  
Shows the state of the sprint ("Planning", "In Progress", "Done").
- *Start on/at*  
Defines the start date of the sprint.
- *End on/at*  
Defines the end date of the sprint.
- *Review on/at*  
Defines the date of the review meeting.
- *Feature Team*  
If feature teams are defined in the scrum project ("Scrum Project" tab > *Feature Teams*), a feature team can be assigned to the sprint.
- *Sprint Topic*  
If sprint topics are defined in the scrum center ("Structuring" tab > *Topics for Structuring Sprints*), a topic can be assigned to the sprint.
- *Commitment (Story Points)*  
Shows the sum of story points of backlog items at the time the sprint was started.
- *Commitment (Count)*  
Shows the number of backlog items at the time the sprint was started.
- *Added After Start (Story Points)*  
Shows the sum of story points of backlog items added after the sprint was started.

- *Added After Start (Count)*  
Shows the number of backlog items added after the sprint was started.
- *Removed After Start (Story Points)*  
Shows the sum of story points of backlog items removed after the sprint was started.
- *Removed After Start (Count)*  
Shows the number of backlog items removed after the sprint was started.
- *Total (Story Points)*  
Shows the sum of story points of all backlog items in the sprint.
- *Total (Count)*  
Shows the number of all backlog items in the sprint.
- *Terms*  
Defines keywords for the sprint.
- *Sprint Goal*  
Defines the goal that should be reached in the sprint.
- *External IDs*  
If the sprint should be processed with external applications, IDs can be defined for each application. The external applications have to be specified in the scrum center or project ("Predefined Values" tab > *External Applications*).

### 6.3.2.2 "History" Tab

The history tab shows events that occurred in the sprint.

### 6.3.2.3 "Capacity Planning" Tab

This tab is only available, if the capacity planning is enabled in the scrum center ("Feature Administration" tab > *Enable Capacity Planning*).

- *Team Availability*  
For each team member the availability in hours can be defined.
- *Planned Capacity*
  - *Projected Total Capacity*  
Shows the sum of the availability of all team members.
  - *Planned Effort*  
Shows the sum of planned efforts of all tasks ("Details" tab > *Planned Effort*).
  - *Remaining Capacity*  
Shows the difference between the projected total capacity and the planned effort.
  - *Time Worked*  
Shows the sum of time worked of all tasks ("Details" tab > *Time Worked in Sprint*).
  - *Estimated Remaining Time*  
Shows the sum of the estimated remaining time of all tasks ("Details" tab > *Estimated Remaining Time*).
  - *Capacity Exceeded by*  
If the planned effort exceeds the projected total capacity the difference is shown in this field.

### 6.3.3 Actions and Context Menu Commands

- **Capacity Planning**  
Creates a capacity plan for a sprint that is in planning or in progress (only available if the capacity planning is enabled in the scrum center). For each team member the available hours can be defined.
- **Create Task**  
Creates a task for a backlog item (only available if at least one backlog item is in the sprint).
- **Start Sprint**  
Starts a sprint that is in planning.
- **End Sprint**  
Ends the sprint that is in progress. A sprint can only be ended, if there are no backlog items with the state "Done" (has to be released or not released). Backlog items that are not completed can be automatically moved to a next sprint or the sprint can just be ended. In this case not completed backlog items get automatically the state "Not Released".
- **Properties**  
Opens the properties of the sprint.

## 6.4 Product Version

In general several versions of a software product will be released (e.g. 2015, 2016). A *Product Version* can be subdivided into *Releases* (e.g. 2015, 2015 SP1). Product versions and releases are useful to track the progress of the stories that are needed to be able to release a new version.

Product versions may be defined in the scrum center and/or in the scrum project.

### 6.4.1 Properties

- *Name*  
The description of the product version.
- *Code Freezed*  
Defines whether the release life cycle has reached the code freeze state.
- *Supported*  
Defines whether the product version is still supported.
- *Terms*  
Defines keywords for the product version.
- *Scrum Centers*  
The same product version can be defined in several scrum centers (copy and paste shortcut). This field shows all scrum centers the product version is assigned to.
- *Scrum Projects*  
The same product version can be defined in several scrum projects (copy and paste shortcut). This field shows all scrum projects the product version is assigned to.
- *External IDs*  
If the product version should be processed with external applications, IDs can be defined for each application. The external applications have to be specified in the scrum center or project ("Predefined Values" tab > *External Applications*).

## 6.4.2 Actions and Context Menu Commands

- **Create Release**  
Creates a new release for a product version.
- **Show Assigned Backlog Items**  
Shows all stories, defects and impediments that are assigned to the product version. The assignment can be changed in the properties of the backlog item ("Details" tab > *Product Version*).
- **Properties**  
Opens the properties of the product version.

## 6.5 Release

In general a product version consists of several releases. Assign the necessary stories to a release to keep track of the work progress.

### 6.5.1 Dashboard

The dashboard is divided into following areas:

- **Story Burndown**  
The burndown is based on the story points of stories with state "Done". The assigned release can be changed in the properties of stories ("Details" tab > *Release*). The story burndown is only available if at least one assigned story exists and a start and end date is defined for the release.
- **Documents**  
Documents that belong to the release can be stored here. Folders can be used to structure the documents.
- **Requested**  
Shows backlog items that are requested for the release ("Story"/"Defect"/"Impediment" tab > *Requested for*).
- **Completed**  
Shows backlog items that are completed with the release ("Story"/"Defect"/"Impediment" tab > *Completed With*).
- **Reported Defects**  
Shows defects that occurred with the release ("Defect" tab > *Occurred With*).
- **Assigned Stories**  
Shows stories that are assigned to the release ("Details" tab > *Release*).
- **Assigned Impediments**  
Shows impediments that are assigned to the release ("Details" tab > *Release*).
- **Assigned Defects**  
Shows defects that are assigned to the release ("Details" tab > *Release*).

### 6.5.2 Properties

- **Name**  
The description of the release.
- **Product Version**  
Shows the superordinate product version.

- *State*  
Shows the state of the release.
- *Start on/at*  
The start date of the release.
- *End on/at*  
The end date of the release.
- *Code Freezed*  
Defines whether the release life cycle has reached the code freeze state.
- *Supported*  
Defines whether the release is still supported.
- *Hotfix Release*  
Defines whether hotfixes are provided for the release. When requesting a hotfix for a defect only those releases can be selected.
- *Terms*  
Defines keywords for the release.
- *External IDs*  
If the release should be processed with external applications, IDs can be defined for each application. The external applications have to be specified in the scrum center or project ("Predefined Values" tab > *External Applications*).

### 6.5.3 Actions and Context Menu Commands

- **Planning**  
Sets the state of the release to "Planning" (only available if the current state is not "Planning").
- **In Progress**  
Sets the state of the release to "In Progress" (only available if the current state is not "In Progress").
- **Done**  
Sets the state of the release to "Done" (only available if the current state is not "Done"). Such releases cannot be selected any longer in backlog items.
- **Change Assignment (only context menu)**  
Releases that are accidentally created for the wrong product version can be reassigned as long as the release is not used in a backlog item.
- **Properties**  
Opens the properties of the release.

## 7 Scrum Center

The scrum center allows defining general settings that apply to all subordinate scrum projects. Additionally, the team members of a scrum center either have special access rights on the scrum projects or can be assigned to the scrum projects as team members.

### 7.1 Dashboard

The dashboard is divided into following areas:

- *Scrum Projects*  
Shows all scrum projects that belong to the scrum center.
- *Product Versions*  
Shows all product versions of the scrum center. Product versions can be assigned to backlog items. Navigate in the widget to create a new product version.
- *Forms and Categories*  
Shows the available forms and categories.  
**Note:** To be able to store forms and categories from a different context, you must adjust the *Restrict Shortcuts Within Teamroom* field in the configuration settings accordingly.
- *Processes*  
Shows the available custom processes.
- *Templates*  
In the create dialog (e.g. "New" context menu command in a Teamroom) the templates are displayed according to the grouping by the template categories.
- *Text Modules*  
The defined text modules can be inserted into Word documents.
- *Report Templates*  
Shows the available report templates. Report templates can be defined as BIRT or Excel reports. The *Scope of Analysis* field defines where the action for generating the report is displayed and which data can be included in the report.
- *Thesauri*  
Shows the thesauri in which terms for tagging of scrum artifacts can be managed. At least one thesaurus has to be created so that the tagging can be used. The default thesaurus for new terms is defined in the settings of the scrum center. Thus, new terms can be created directly in the tagging process. If no standard thesaurus is specified, only the predefined terms in the thesauri can be used.

## 7.2 Settings

A scrum center provides following settings.

### 7.2.1 "General Settings" Tab

- *Name*  
The name of the configuration.
- *Subject*  
Defines an additional description of the configuration.
- *Mindbreeze InSpire Service*  
Defines the Mindbreeze InSpire Service that is used for classification.
- *Access Protection*  
Defines the access protection of the scrum center.
  - Default: Only the defined scrum center members can access the scrum center and its content.
  - Generally readable: Everyone can read the scrum center and its contents, but searching is not possible.

- Restricted: Only scrum administrators and privileged users have access to the members of the scrum center. Typically, this setting is used, if the scrum center contains several scrum projects of different partners. This way the partners cannot see the users who are defined in the scrum center.
- *Restrict Shortcuts Within Teamroom*  
Defines which type of shortcuts may be stored in the configuration.
- *Restrict the Downloading or Opening of Content on the Device*  
Allows to restrict team members who can open or download content at the device.
- *All Team Members May Add Members*  
Defines whether all team members can add users to the team or only team members with full control.
- *Restrict Team Members*  
Defines the organizations, organizational units, teams and external organizations whose members may be added to the configuration.
- *Enable Advanced Mode*  
Advanced mode includes, for example, working with multiple shelves, as well as managing templates and presettings.
- *Main Administrator*  
Defines the user who will receive the automatically generated e-mail messages concerning the configuration. Otherwise, all scrum center administrators receive the e-mail messages. The user is also listed as contact in case of missing permissions.
- *Support Team*  
The support team handles the organization-internal management of support requests in the respective context.
- *Activate Trace Output*  
For example, if you are a form designer using expressions for calculation or validation, it can sometimes be difficult to identify errors in the expressions. To simplify analysis, you can write trace output to the web browser console `coobj.Trace("string");` or `coobj.Trace("string", value);`.
- *Organization*  
Shows the organization the scrum center is assigned to.

## 7.2.2 "Scrum Center" Tab

- *Short Form*  
The short form is used as prefix for the backlog item ID. If left empty, it is enforced that each scrum project defines its own short form.
- *Take Over Product Versions From Scrum Projects*  
Defines whether product versions that are created in scrum projects are automatically added to the scrum center. This applies only to product versions created after enabling the setting.
- *Use Product Versions Only From the Scrum Center*  
Defines whether only product versions that are defined in the scrum center can be used in backlog items. Specific scrum projects can be excluded.
- *Projects With Own Product Version Management*  
Defines projects that may specify their own product versions, if *Use Product Versions Only From the Scrum Center* is correspondingly configured.

- *Allow Story Points for Defects and Impediments*  
Enables the *Story Points* field also for defects and impediments.
- *Prevent Changes of Closed Backlog Items*  
Defines whether the contents of closed backlog items can be changed.
- *Require Unique ID per External Application*  
Defines whether the ID for external applications has to be unique.

### 7.2.3 “Feature Administration” Tab

- *Enable Capacity Planning*  
Defines whether the capacity planning is enabled. If enabled, the effort for each task can be defined. Additionally, the amount of available hours of each team member can be specified per sprint.
- *Unit for Effort Tracking*  
If the capacity planning is enabled, the unit for effort tracking is “hours”. If disabled, it is set to “man days”.
- *Enable Defects Reports*  
Allows generating a daily defects report as PDF. In the *Report Settings* you can for example specify the generation time, the included scrum projects and the recipients.

### 7.2.4 “Predefined Values” Tab

- *Processing Stages*  
A processing stage (e.g. “Implementation”, “Quality Assurance”, “Documentation”) can be assigned to a task. If all tasks of a processing stage are done the story will turn to the next processing stage. The order of the processing stages is defined by the order in this field.
- *Defect Severities*  
Defects may be classified by their severities.
- *Affected Applications*  
Affected applications may be assigned to backlog items.
- *External Applications*  
Defines a set of external applications. Scrum artifacts can define an ID for these applications. Based on this ID external applications can process the scrum artifacts.
- *Environments*  
Environments (e.g. Linux) may be assigned to backlog items.
- *Impact Areas*  
Impact areas (e.g. GUI) may be assigned to backlog items.
- *Hierarchy Levels*  
Hierarchy levels (e.g. software layers) may be assigned to backlog items.
- *Administrative Categories*  
Administrative categories may be assigned to backlog items.

### 7.2.5 “Structuring” Tab

The defined terms in the following fields can be used to structure the corresponding scrum artifacts.

- *Topics for Structuring Backlog Items*

- *Topics for Structuring Tasks*
- *Topics for Structuring Sprints*
- *Topics for Structuring Time Tracking Entries*
- *Topics for Structuring Reasons of Defects*
- *Topics for Structuring Causes of Defects*

### 7.2.6 “Tagging” Tab

The defined terms in the following fields can be used to tag the corresponding scrum artifacts.

- *Terms for Tagging Stories*
- *Terms for Tagging Defects*
- *Terms for Tagging Impediments*
- *Terms for Tagging Tasks*
- *Terms for Tagging Sprints*
- *Terms for Tagging Versions*
- *Terms for Tagging Releases*

### 7.2.7 “User Interface” Tab

- *Prevent Time Entry Editing*  
Defines whether time entries can be changed later on.
- *Additional Required Fields*  
Defines fields that are mandatory.
- *Additional Locked Fields*  
Defines fields that cannot be changed.
- *Enable Checkboxes for the Following Fields*  
Defines fields that only allow selecting predefined values.
- *Allow Single Selection Only*  
Fields defined in *Enable Checkboxes for the Following Fields* can be restricted to allow selecting only one value.

### 7.2.8 „Logos” Tab

The defined logos and background image are used for the scrum center and for all personal dashboards.

**Note:** If the personal dashboard is assigned to several configurations, the logos and background image will not be applied.

### 7.2.9 “Default Values” tab

- *Default Thesaurus for New Terms*  
If a default thesaurus is specified, new terms can be created directly during tagging.
- *Default Categories*  
When an object is created, the specified category is assigned according to the defined object class.

- *Default Processes*  
When an object is created, the specified process is assigned according to the defined object class.
- *Default Background Tasks*  
When an object is created, the specified background tasks are assigned.

## 7.3 Actions

- **Create Scrum Project**  
Creates a new scrum project. The settings in the scrum center will be inherited by the scrum project.
- **Translations**  
The "Translations" action can be used to export or import a CSV file containing the multilingual names of the customizing objects. This enables an external translation.
- **Authorize Team**  
Defines the users of the scrum center. A description of the different roles can be found in chapter 4 "Roles".
- **Settings**  
Opens the settings of the scrum center.

## 8 Scrum Project

Each scrum project is assigned to a scrum center. The scrum project defines the scrum team and provides a central repository for all scrum artifacts.

### 8.1 Dashboard

The dashboard is divided into following areas:

- *Velocity*  
Shows the performance of the team in the last sprints based on story points. The diagram is only visible if at least two sprints have ended.
- *Defects*  
Shows all open defects.
- *Current Stories and Defects*  
Shows all stories of currently running sprints and all open defects.
- *Inbox*  
Shows new stories that are not yet accepted or rejected.
- *Backlog*  
Shows accepted stories. Accepted stories can be added to sprints.
- *Sprints*  
Shows all sprints that are not archived. To view the archived sprints, open the sprint widget and click the "Open Sprint Archive" action (only available, if at least one archived sprint exists).
- *Impediments*  
Shows all open impediments.

- *Documents*  
Documents that belong to the scrum project can be stored here. Folders can be used to structure the documents.
- *Product Versions*  
Shows all product versions of the scrum project. Product versions can be assigned to backlog items. This widget is only available if the setting in the scrum center allows product versions for this scrum project.
- *Backlog Done*  
Shows all not archived stories that are released or rejected. To view the archived backlog items, open the backlog done widget and click the “Open Backlog Archive” action (only available, if at least one archived backlog item exists).

## 8.2 Settings

A scrum project provides following settings.

### 8.2.1 “General Settings” Tab

- *Name*  
The description of the scrum project.
- *Subject*  
Defines an additional description of the scrum project.
- *Enable Notification*  
Defines whether you are notified about new events of this scrum project.
- *Mindbreeze InSpire Service*  
Defines the Mindbreeze InSpire Service that is used for classification.
- *Access Protection*  
Defines the access protection of the scrum project.
  - Default: Only the defined scrum center team and scrum project team can access the scrum project and its contents.
  - Generally readable: Everyone can read the scrum project and its contents, but searching is not possible.
  - Restricted: Only the scrum project team, privileged users (scrum center) and scrum administrators (scrum center) can access the scrum project and its contents. Typically, this setting is used, if a scrum center contains several scrum projects of different partners. This way scrum users and stakeholders defined in the scrum center cannot access the scrum project.
- *Restrict Shortcuts Within Teamroom*  
Defines which type of shortcuts may be stored in the scrum project.
- *Restrict the Downloading or Opening of Content on the Device*  
Allows to restrict team members who can open or download content at the device.
- *Restrict Team Members*  
Defines the organizations, organizational units, teams and external organizations whose members may be added to the scrum project.
- *Main Administrator*  
Defines the user who will receive the automatically generated e-mail messages concerning the

shelf. Otherwise, all scrum project administrators receive the e-mail messages. The user is also listed as contact in case of missing permissions.

- *Support Team*  
The support team handles the organization-internal management of support requests in the respective context.

### 8.2.2 “Scrum Project” Tab

- *Inbox*  
Shows the inbox that is assigned to the scrum project.
- *Vision Document*  
Defines a document that summarizes the objective of the scrum project.
- *Scrum Center*  
Shows the scrum center the scrum project is assigned to.
- *Feature Project*  
Defines a superordinate scrum project. If a feature project is defined, only stories from the feature project can be selected as feature in backlog items.
- *Project Specific Numbering*  
Defines whether the scrum project defines its own short form. Otherwise the short form of the scrum center is used.
- *Short Form*  
The short form is used as prefix for the backlog item ID.
- *State*  
No longer used scrum projects can be “Closed”. For example no new defects or stories can be assigned to closed scrum projects.
- *Velocity*  
Shows the average of story points that are released per sprint.
- *Enable Detail Planning for Tasks*  
Defines whether tasks and efforts can be defined when adding a story to a sprint. The option is only available if the capacity planning is enabled in the scrum center.
- *Show Hours Burndown*  
Enables the hours burndown widget for sprints. The option is only available if the capacity planning is enabled in the scrum center.
- *Show Story Burndown*  
Enables the story burndown widget for sprints. The burndown is based on the story points of stories with state “Done”.
- *Show Task Burndown*  
Enables the task burndown widget for sprints. The burndown is based on the amount of tasks with state “Done”.
- *Send E-Mail Notifications*  
Defines whether e-mail notifications are sent. For example, an e-mail is sent to the product owner when a new story is added to the inbox. Users who requested backlog items will be informed about state transitions. If a hotfix is requested the whole team gets an e-mail. In addition, if a task is assigned to a user or taken away from a user also an e-mail will be sent.
- *Prefix for Subject of E-Mail Notifications*  
Defines the prefix string that is added to the subject of notification e-mails.

- *Archive Sprints After*  
Sprints are archived after the defined time span (from one month to three years).
- *Archive Items in "Backlog Done" After*  
Items in "Backlog Done" are archived after the defined time span (from one month to three years).
- *Maximum Entries in "Backlog Done"*  
Backlog items in "Backlog Done" are archived, if the defined number is exceeded. The number also defines the maximum elements per archive. If the number is exceeded a new archive will be created.  
**Note:** Only one condition either *Archive Items in "Backlog Done" After* or *Maximum Entries in "Backlog Done"* must be fulfilled such that the backlog item is archived.
- *Maximum Number of Versions Kept*  
Defines how many versions should be kept at maximum. When exceeding the limit, the oldest versions are deleted.
- *Sprint Overview*  
Shows all ended but not archived sprints.
- *Feature Teams*  
Defines feature teams, if several teams work in one scrum project.

### 8.2.3 "Initialization" Tab

In the following fields several default values can be defined for backlog items.

- *Default Text for Story*
- *Default Acceptance Criteria for Story*
- *Default Tasks for New Story*
- *Default Tasks for Story Added to Sprint*
- *Default Text for Defect*
- *Default Tasks for New Defect*
- *Default Tasks for Defect Added to Sprint*
- *Default Text for Impediment*
- *Default Tasks for New Impediment*
- *Default Tasks for Impediment Added to Sprint*

### 8.2.4 "Predefined Values" Tab

- *Processing Stages*  
A processing stage (e.g. "Implementation", "Quality Assurance", "Documentation") can be assigned to a task. If all tasks of a processing stage are done the story will turn to the next processing stage. The order of the processing stages is defined by the order in this field.
- *Defect Severities*  
Defects may be classified by their severities.
- *Environments*  
Environments (e.g. Linux) may be assigned to backlog items.
- *Affected Applications*  
Affected applications may be assigned to backlog items.

- *External Applications*  
Defines a set of external applications. Scrum artifacts can define an ID for these applications. Based on this ID external applications can process the scrum artifacts.
- *Impact Areas*  
Impact areas (e.g. GUI) may be assigned to backlog items.
- *Hierarchy Levels*  
Hierarchy levels (e.g. software layers) may be assigned to backlog items.
- *Administrative Categories*  
Administrative categories may be assigned to backlog items.

### 8.2.5 “Structuring” Tab

The defined terms in the following fields can be used to structure the corresponding scrum artifacts.

- *Topics for Structuring Backlog Items*
- *Topics for Structuring Tasks*
- *Topics for Structuring Sprints*
- *Topics for Structuring Time Tracking Entries*
- *Topics for Structuring Reasons of Defects*
- *Topics for Structuring Causes of Defects*

### 8.2.6 “Tagging” Tab

The defined terms in the following fields can be used to tag the corresponding scrum artifacts.

- *Terms for Tagging Stories*
- *Terms for Tagging Defects*
- *Terms for Tagging Impediments*
- *Terms for Tagging Tasks*
- *Terms for Tagging Sprints*
- *Terms for Tagging Versions*
- *Terms for Tagging Releases*

### 8.2.7 “Default Values” tab

- *Default Thesaurus for New Terms*  
If a default thesaurus is specified, new terms can be created directly during tagging.
- *Default Categories*  
When an object is created, the specified category is assigned according to the defined object class.
- *Additional Categories*  
Defines categories in addition to the categories defined in the configuration.
- *Default Processes*  
When an object is created, the specified process is assigned according to the defined object class.

- *Additional Default Background Tasks*

When an object is created, the specified background tasks are assigned.

**Note:** The default properties overwrite or extend the settings in the configuration.

### 8.3 Actions

- **Create Story**  
Creates a new story.
- **Create Defect**  
Creates a new defect.
- **Create Impediment**  
Creates a new impediment.
- **Create Sprint**  
Creates a new sprint.
- **Show New Events**  
Shows events that occurred in the scrum project.
- **Open Wastebasket**  
Opens the wastebasket (only available if at least one object is in the wastebasket).
- **Templates and Presettings**  
Allows to define forms and categories, process templates, templates, text modules, report templates and presettings.
- **Settings**  
Opens the settings of the scrum project.
- **Team**  
Allows to define or view the team members of the scrum project. A description of the different roles can be found in chapter 4 "Roles".

## 9 Additional Tasks

For an easy migration of backlog items from other systems a CSV import can be carried out by product owners.

To import backlog items, perform the following steps:

1. On your personnel dashboard or on a scrum project execute the "Import Backlog Items" context menu command.
2. Enter the path to the CSV file in the *Content* field.  
**Note:** Click the "Download CSV Template" button to retrieve a template that describes the necessary data structure.
3. Click "Start Import".
4. After the import has finished, click "Next".

**Data structure of the CSV file**

CSV Column	Description
------------	-------------

Type	the type of the backlog item ( <i>Story</i> , <i>Defect</i> , <i>Impediment</i> )
ImportID	an arbitrary unique key to identify the item in recurring imports (string)
ScrumProjectID	the target scrum project (Fabasoft Cloud ID)
Title	the <i>Subject</i> of the backlog item (string)
Feature	the <i>Feature</i> of the backlog item (Fabasoft Cloud ID)
State	the <i>State</i> of the backlog item ( <i>Received</i> , <i>Planning</i> )
Priority	the <i>Priority</i> of the backlog item ( <i>SPT_A_MUST_HAVE</i> , <i>SPT_B_SHOULD_HAVE</i> , <i>SPT_C_COULD_HAVE</i> , <i>SPT_D_WONT_HAVE</i> )
Order	the <i>Order</i> of the backlog item (integer)
StoryPoints	the <i>Size</i> of the backlog item ( <i>F_0</i> , <i>F_1</i> , <i>F_2</i> , <i>F_3</i> , <i>F_5</i> , <i>F_8</i> , <i>F_13</i> , <i>F_21</i> , <i>F_34</i> , <i>F_UNDEFINED</i> )
Story	the description of the backlog item (quoted string)
AcceptanceCriteria	the <i>Acceptance Criteria</i> of the backlog item (quoted string)
CreatedAt	<i>Created on/at</i> of the backlog item (format: yyyy-mm-dd hh:mm:ss)

**Note:**

- Backlog items can be updated during a new import, but only if the state has not changed in the meantime.
- When importing via the personnel dashboard the scrum project ID has to be specified, if multiple scrum projects exist.
- When importing via a scrum project only entries are imported whose scrum project IDs are either empty or correspond to the current project.

## 10 Administration Tasks

In the following chapters you will find administration tasks that can be carried out by scrum administrators and partly by privileged users.

- A scrum administrator can administer and define the team members and security settings of the scrum center and all assigned scrum projects. Additionally, a scrum administrator can create new scrum centers.
- A privileged user can administer the scrum center and all assigned scrum projects, but is not allowed to define the team members or security settings.

## 10.1 Prerequisites

To be able to use Fabasoft Scrum, the corresponding app has to be assigned to your organization. This can be checked in the properties of the organization ("Service Packages" tab > *Editions and Apps*). If it is not assigned, please contact Fabasoft Sales.

After the app has been assigned to the organization the organization owner, payment user and organization administrators receive a welcome screen notification that a scrum center must be created.

You can forward the notification to a responsible user or you can create a scrum center right away. After creating a scrum center, the scrum dashboard is put on "Home" of the specified scrum administrators and the notification will be removed from all receivers.

## 10.2 Scrum (Personal Dashboard)

The personal dashboard provides two actions for scrum administrators:

- Create Scrum Center  
Creates an additional scrum center. This is for example necessary, if scrum projects need different scrum center settings.
- Create Scrum Project  
Creates a scrum project in a scrum center.  
**Note:** This action is also available for privileged users of a scrum center.

### 10.2.1 Scrum Center

The scrum center allows defining general settings that apply to all subordinate scrum projects. The team members of a scrum center either have special access rights on the scrum projects or can be assigned to the scrum projects as team members.

**Note:** On the basis of the defined users the consumed licenses are calculated.

#### 10.2.1.1 Change the Settings

To change the settings of a scrum center, perform the following steps:

5. Navigate in the scrum center (via the scrum dashboard).
6. Click the "Settings" action.
7. Change the settings as desired. The possible settings of a scrum center are described in chapter 7 "Scrum Center".
8. Click "Next".

#### 10.2.1.2 Authorize the Team

To authorize the team of a scrum center, perform the following steps:

1. Navigate in the scrum center (via the scrum dashboard).
2. Click the “Authorize Team” action.
3. Define the team members as desired. The different roles are described in chapter 4 “Roles”.

## 10.2.2 Scrum Project

Each scrum project is assigned to a scrum center. The scrum project defines the scrum team and provides a central repository for all scrum artifacts.

### 10.2.2.1 Change the Settings

1. Navigate in the scrum project (via the scrum dashboard).
2. Click the “Settings” action.
3. Change the settings as desired. The possible settings of a scrum project are described in chapter 8 “Scrum Project”.
4. Click “Next”.

### 10.2.2.2 Authorize the Team

To authorize the team of a scrum project, perform the following steps:

1. Navigate in the scrum project (via the scrum dashboard).
2. Click the “Authorize Team” action.
3. Define the team members as desired. The different roles are described in chapter 4 “Roles”.

## 11 Web Service

The scrum web service can be used by third-party applications to access scrum projects. The web service is invoked in context of a user. The authentication is carried out with a “password for applications” that is managed by the user.

To define a password for applications, perform the following steps:

1. In the Fabasoft Cloud Web Client open the account menu (your user name).
2. Click “Access for Applications”.
3. In the *Passwords for Applications* field click “New”.
4. In the *Valid for* field select “Scrum Projects”.
5. Define a *Validity Period* and click “Next”.
6. Copy the displayed password that can be used for the authentication of the web service.
7. Click “Close”.

### 11.1 SOAP and WSDL

You can use SOAP to call the operations listed in the scrum WSDL.

The WSDL can be retrieved via following URL:

`https://<host>/<vdir>/fscdav/wsd1?WEBSVC=FSCSCRUM_1_1001_ScrumWebService`

**Example:**

[https://at.cloud.fabasoft.com/folio/fscdav/wsdI?WEBSVC=FSCSCRUM\\_1\\_1001\\_ScrumWebService](https://at.cloud.fabasoft.com/folio/fscdav/wsdI?WEBSVC=FSCSCRUM_1_1001_ScrumWebService)

## 11.2 JSON

The scrum web service also supports the JSON format. You can call the operations the following way:

`https://<host>/<vdir>/wsjson/FSCSCRUM_1_1001_ScrumWebService/<operation>`

**Example:**

[https://at.cloud.fabasoft.com/folio/wsjson/FSCSCRUM\\_1\\_1001\\_ScrumWebService/GetScrumProjects?all=true](https://at.cloud.fabasoft.com/folio/wsjson/FSCSCRUM_1_1001_ScrumWebService/GetScrumProjects?all=true)

## 11.3 Available Operations

The scrum web service provides following operations:

- AddStoryRemark
- CreateScrumDefect
- CreateScrumImpediment
- CreateScrumStory
- CreateScrumTask
- GetMultipleScrumProjects
- GetMultipleScrumReleases
- GetMultipleScrumRequestContainers
- GetMultipleScrumSprints
- GetMultipleScrumStories
- GetMultipleScrumTasks
- GetMultipleScrumVersions
- GetScrumChildren
- GetScrumObjectInfo
- GetScrumProject
- GetScrumProjectRejectedStories
- GetScrumProjectReleasedStories
- GetScrumProjects
- GetScrumProjectStories
- GetScrumProjectVersions
- GetScrumRelease
- GetScrumReleaseStories
- GetScrumRequestContainer
- GetScrumSprint
- GetScrumSprintStories
- GetScrumStory
- GetScrumStoryChanges
- GetScrumStoryDetails
- GetScrumStoryHistory
- GetScrumStoryRepositoryInfo

- GetScrumStoryTasks
- GetScrumTask
- GetScrumTaskHistory
- GetScrumTracklist
- GetScrumVersion
- GetScrumVersionReleases
- GetStoryRemarks
- RequestHotfix
- SearchScrumProjectStories
- SearchScrumProjectTasks
- UpdateScrumStory
- UpdateScrumTask

More information about the parameters of the operations can be found here:

<https://help.appducx.com/index.php>. Search for the corresponding actions. Generally, the actions

assigned to the operations have the same names. **Exceptions:** GetScrumProject (GetScrumProjectInfo), GetScrumStoryDetails (GetScrumStoryExtendedDetails), RequestHotfix (RequestScrumHotfix)