



Sirid User's Guide

Table of contents

- 1. System configuration (for admins).....3**
 - 1.1. Email server and account.....3
 - 1.2. Localization settings.....3
 - 1.3. Responsible roles.....3
 - 1.4. Staff member templates.....4
 - 1.5. Task and bug categories.....4
 - 1.6. Getting updates.....4

- 2. Setting up users and projects (for admins).....5**
 - 2.1. Users.....5
 - 2.1.1. User groups.....5
 - 2.2. Project folders.....6
 - 2.3. Projects.....6
 - 2.4. Staff members.....6

- 3. Specification phase (for project leaders / architects).....8**
 - 3.1. Specifications.....8
 - 3.2. Tasks.....8

- 4. Implementation and testing phases (for all users).....9**
 - 4.1. Viewing specifications, tasks and bugs.....9
 - 4.2. Implementing tasks and fixing bugs.....9
 - 4.3. Reporting bugs.....10
 - 4.4. Timesheet.....10

- 5. Miscellaneous activities (for all users).....12**
 - 5.1. Feature requests.....12
 - 5.2. Personal settings.....12
 - 5.3. Viewing statistics.....12
 - 5.4. Tracking changes.....12

1. System configuration (for admins)

The first step after installing Sirid is usually modifying the configuration to match your needs. This step is optional if you want to use the default values. All of these settings can be modified at any time by an administrator user.

1.1. Email server and account

The email server and account are already configured in the installation but if you did not specify these settings or you want to modify them, they are available under the Settings menu. Sirid uses the given email information to send various notification messages. The email server must allow sending messages from the computer you install Sirid to, so you typically want to specify the mail server of your Internet provider or the mail server of your company.

1.2. Localization settings

Sirid allows you to specify various settings that make its usage more convenient in your country. All of the localization settings are available under the Settings menu. Date format specifies the date format you want to use to input and output dates. Default language affects the language of the login page as well as the default language for new users. Timezone can be useful if the server is located in a different place than your company but there is normally no need to modify this setting. First day of week specifies the starting day of a week and is used in all calendars.

1.3. Responsible roles

Responsible roles are meant to distinguish multiple responsible persons for a single task or bug. The responsible role specifies what kind of changes to the entry should result in notification message to be sent to the responsible and whether the entry is shown in the "My tasks" / "My bugs" page of the responsible person. Naturally the name of the role makes the clearest indication of the type of the role: "Head responsible" is responsible for the task to be completed while "Assistant" might be responsible for doing some part of the task or providing some information and a person with "Keep informed" role might not be related to implementation but just needs to be notified when the job is done.

By default Sirid contains the three roles mentioned above; Head responsible, Assistant and Keep informed. You can freely modify all responsible roles apart from the Head responsible role, which is a built-in role with built-in attributes. Responsible roles can be modified via the Miscellaneous menu.

1.4. Staff member templates

Staff member templates are predefined types for project members. Staff member template specifies the job description of the person and default rights for that job. Typical template names are such as "Project leader" and "Programmer". The former template could for example have the "Manage project" access right while the latter usually would not. The template values are only defaults and all values can be modified when adding a user to a project.

By default Sirid contains five Staff member templates: Architect, Chief programmer, Programmer, Project leader and Tester. Position templates can be modified via the User maintenance menu.

1.5. Task and bug categories

Task and bug categories are used to group tasks or bugs that are somehow similar to each other. Sirid has two types of categories: Global categories and project specific categories. The project specific categories are only visible when adding an entry to that specific project while the global categories are visible for all projects.

The categories can be used in many very different ways. One way is to use the categories to indicate a general area in the project, such as "User interface" or "Algorithm". Another way is to indicate the type of the entry, like "New feature", "Optimization", "Dead code", etc. Third way is to use the project specific categories and use the categories to represent various components in a project, like "Native connector", "Graphic renderer", "Parser engine", etc. (with different categories for each project).

By default Sirid contains four categories (same for both tasks and bugs): Algorithm, Graphics, Install/Uninstall and User interface. The categories can be modified via the Miscellaneous menu.

1.6. Getting updates

Sirid can be updated very easily by just clicking the "Look for updates" link in the Maintenance menu. If an update is available Sirid will show your current version, the new version and provides a link to version history from which you can see what has changed. After confirming the update Sirid will download and install the latest version. Normally you only need to re-login to the system but if you experience any problems you should restart the server.

2. Setting up users and projects (for admins)

Users and projects are the two most fundamental concepts in Sirid. Before you can start using Sirid you must add some users and projects to the system, and users to the projects. It is important to notice that a (non-administrator) user cannot do practically anything before he or she is assigned to one or more projects.

2.1. Users

New users can be created via the User management menu. Each user has five very important fields: Initial password, login name, user type, user name and email address. Other fields are less important and whether you fill them or not does not have a major impact on the system usability.

Initial password specifies the password the user can use to login to the system for the first time, and which the user should then immediately change. Login name is the name the user uses to login to the system. There is no restriction to the format of the login name but we recommend either first letter of first name and full last name, or full first name and first letter of last name. That is, for John Doe, either `jdoe` or `johnd`. Login name is thus the "writer-friendly" name of the user, while User name is the "reader-friendly" version, that is, the full real name of the user. User type specifies whether the user is allowed to do administrative tasks (tasks listed in chapters 1 and 2). There is no restriction to the amount of administrator users but it is recommended to have at most "a few" administrators. Email address specifies the address of the user where Sirid should send messages when some action concerning that user is taken.

2.1.1. User groups

User groups can be accessed via the user management menu (Manage user groups). User groups specify sets of users that have limited access to the projects. In practice the limited access means that members of user groups should generally see only tasks and bugs that are related to a member of the group. You could, for example have a group named "Subcontractor A" or a group named "Customers of project B". The members of this kind of group should usually see only a limited list of tasks and bugs. In the case of "Subcontractor A", the members of that group should only see tasks and bugs some member of that group is responsible for, and in the case of "Customers of project B", the members of the group should see only such bugs that have been added by a member of the group.

Sirid automatically takes care of showing only the appropriate tasks and bugs for users if you specify limited access rights when adding the user to the project. See section 2.4, Staff members for more information regarding staff member management.

2.2. Project folders

Project folders provide a way to categorize your projects. Using project folders is optional, while recommended if you have about ten or more projects. Typically you want the folder hierarchy to reflect different departments in your company or different types of projects. The first level of folders could contain such folders as Internal projects, Customized projects and Retail projects. These folders could further be divided into other subfolders if necessary.

2.3. Projects

Administrator users can add new projects via the Projects menu. Each project has only a few required attributes (such as name and folder) but in order for the project to be meaningful it must have also at least one version and some project members (staff). Most entries in Sirid are tied to a project or to a specific version of the project. For example tasks and bugs are tied to a specific project version while (currently) project specifications are tied to the project itself.

Administrators and users with the "Manage project"—access right can get to the Project management menu via the Projects page (Management). The Project management menu provides access to the various attributes of the project, such as basic project information, project versions, important dates, staff members, other access rights and project specific categories.

Project versions should specify clearly separate versions of the project. How "clearly separate" is defined depends on a project and your needs and it is difficult to give any clear guidelines of when one should add new version. Each version may have a starting date and deadline which specify the timeframe when the version should be implemented.

Important dates are currently meant to be like the CVS tags. For example, when you release a new minor version of your project (for which you do not create a new version in the system), you could add an important date stamp to indicate when that version was released. You can use these stamps for example when tracking changes between different release versions of your product.

Adding employees and other access rights is described in the next subchapter.

2.4. Staff members

Before a non-administrator user can see a project or make any modifications to it, the user must be assigned to the project and given the appropriate access rights. Users can be added to projects via the Project management menu (accessible via Projects menu [Management]).

Each staff member has a job description, status and a set of access rights. Job description should describe as well as possible what the user is supposed to do for the project. The access rights are the most important property of a staff member. They specify what the user can see about the project and what modifications he or she is allowed to do. The access rights assigned to the user depend



on the role of the user. Project manager might have all access rights while a tester might be able only to view bugs and tasks and report new bugs.

In addition to staff members, the project also has "Other access rights" property. With other access rights you can make the project visible also to people other than the staff members. The most fundamental difference between Staff members and Other access rights is that only Staff members can be set responsible for tasks and bugs. You generally want to give these access rights to your customers so that they can directly report bugs and add feature requests to the system.

3. Specification phase (for project leaders / architects)

Before the actual implementation of a project can begin the project must have specifications, the project must be divided into smaller tasks and these tasks must be assigned to proper people.

3.1. Specifications

The specification system in Sirid mainly tries to provide one single place where to put all specifications and where all project members can see the up-to-date versions of the specifications, removing the hassle created by the dozens of different documents normally emailed to all project members. The specification page is available via the project main page (Projects → Project name).

The specifications are divided into two groups: Project notes and attachments. Project notes can be any (html formatted) text. Sirid naturally keeps track of the modification dates of the notes so everyone can see when, and by whom, the note was last modified. The project notes should generally be fairly short, 1000–2000 words or so, because reading documents longer than this can be tedious due to the restricted format of the texts. Longer texts should be made using some more advanced editor (such as MS Word) and stored as attachments.

Attachments can be any kind of files. Typical examples are Word documents, UML diagrams, Layout examples, and so on. Sirid allows you to specify versions for the attachments and keeps track of attachment history. If the files are already available on some other web server you can also, instead of uploading the attachment to the Sirid server, directly link to the "external attachment".

3.2. Tasks

After or while writing the specifications the project should be divided into smaller parts (2–10 h) and these parts, tasks, should be assigned to some staff members. New tasks can be added by clicking the Add new task link in the main menu. The tasks can naturally be modified later on and new tasks can be added at any time but successful projects are usually well defined from day one.

Most important properties of a task are its title which should be elaborative enough but still not too long, its description which should be covering enough, and the responsible person or persons. A task should usually have only one head responsible because often "shared responsibility is the same as no responsibility at all." For more information regarding responsible roles, see chapter chapter 1.3, Responsible roles.

After adding the task you can also add attachments to the task via the View task page. The attachments can again be of any type, usually being something similar to the attachments in the project specification. I.e. Word documents, UML diagrams, Layout examples, etc.

4. Implementation and testing phases (for all users)

The implementation and testing phases cover the biggest part of the project lifecycle and concern all users. Typical activities in the implementation and testing phases are reading specifications, implementing tasks, reporting bugs and fixing bugs.

4.1. Viewing specifications, tasks and bugs

Before starting any tasks all project members should have a grasp of what the project is about. This information is gained by attending to meetings and by reading the specifications of the project, stored in Sirid. The specifications are available, quite presumably, on the project Specifications page.

There are various different ways of navigating to places of interest. One generally easy way is to use the quick menu on top of the page; simply select the project and page of interest and press Go. Another typical way is to go via the Projects menu.

From normal user's point of view the system consists of two types of tasks and bugs: Those that you're responsible for and those you're not. The tasks and bugs you are responsible for can be seen on My tasks and My bugs pages. Other tasks and bugs can be seen on the appropriate project's Tasks and Bugs page. More detailed information of any entry can be seen by clicking on its title.

4.2. Implementing tasks and fixing bugs

When a task is being implemented or a bug is being fixed its status is usually changed multiple times from one state to another. The status of a task can be changed via the View task page and the status of a bug respectively from the View bug page. Typical phases of a task implementation could be like this:

1. Mary starts implementing a task created by Jack. Mary changes the status of the task to "In implementation" and possibly writes some comment.
2. Mary finishes the task and in her opinion it seems to work as required. She changes the status to "In testing", specifies the amount of time used for the implementation, and writes some comment about what she has done regarding the feature. She also sets Jack as the head responsible of the task and chooses that an email notification of the status change should be sent to the responsible persons.
3. Jack gets a notification that the task is now in testing and that he's head responsible for it. He tests how the system works and notices that something does not work as he had intended. He changes the task status back to "In implementation" and writes a comment in which he explains what part of the task was not correct. He also changes Mary back to the head responsible and

chooses to send an email notification to the responsible persons.

4. Mary gets a notification that the task is back in implementation phase and reads Jack's comments regarding what was done wrong or wasn't done at all. She fixes the parts issued by Jack and does the same as in step 2.

5. Jack gets again a notification that the task is in testing. He checks how the system works and sees that everything works fine this time. He then just changes the status to Done which causes the task to be omitted from the My tasks list of any responsible persons and there is no need to do anything further regarding the task, unless it is re-opened later on.

Same kind of process goes for bugs. Of course, depending on the project and the operation mode of your company, the process needs not be this formal and some steps can be left out.

4.3. Reporting bugs

When you find a bug in a project you have "Report bugs"-access right to, you can press the "Report new bug"-link to report the bug. You should, however, first verify that the bug has not already been reported. You can do this by either reading through the (titles of the) bugs in the project or by using the search functionality to search for bugs that might address the same bug you're about to report.

It is important to fill all fields with care for the responsible person to be able to understand how exactly the bug comes up. The title is naturally essential – it should immediately give a grasp of the bug while being concise enough. The description and how-to-repeat fields are none the less important. Fixing a bug is usually dramatically easier when the responsible person can reproduce it. If there is no simple way of reproducing the bug the description should be elaborative enough for the responsible person to make an educated guess of what could cause the problem.

Regarding the responsible persons, the same rule applies as with tasks: There should be only one head responsible because often "shared responsibility is the same as no responsibility at all." If you do not know who should fix the problem, you should first try to determine that by the job descriptions of the staff members and if you are still uncertain choose the project leader or equivalent who can then "forward" the bug to the correct person. For more information regarding responsible roles, see chapter 1.3, Responsible roles.

After you have reported the bug you can add attachments to it via the View bug page (providing you have the "Add attachments"-access right to the project). The attachments for bugs are usually log files and screenshots but the format of the attachments is not limited in any way.

4.4. Timesheet

The timesheet is a place where you can centrally record your efforts and see what you have done in the past. On company level the timesheet functionality can be used to assist in billing the customers and calculating the salaries of employees.

By clicking a date in the calendar, you can see your efforts on that day. Pressing the Add time link

will add an entry to the selected day, thus allowing you to add entries also to the past when necessary. In addition to adding task and bug related time (change task status and change bug status), you can add project related time. Project related time should be used when your work is not related to any task or bug in the project.

The time tracking functionality enables you to search your time usage in the past. Normal users can only track their own efforts while administrators and accountants can also search the time usage of other users.

5. Miscellaneous activities (for all users)

This section handles features that are not clearly a part of any phase of the project's lifecycle but are still none the less important.

5.1. Feature requests

In order to prevent the project from falling apart it is important that only certain people are allowed to add new tasks. People other than these can, however, come up with really good ideas for the project and there needs to be a way they can share their ideas with the people who decide what to do. For this purpose there are feature requests. Feature requests can be viewed and added via the Project main page. A feature request entry as such is very simple. It only contains the title of the request, description of the request, and the person who should decide whether this is a valuable idea or not.

5.2. Personal settings

Personal settings are related mostly to user interface issues and default values. Most important part of the personal settings is the column customization feature which enables you to choose which information to show on each page and some other related attributes. You should go through your personal settings to see that the values are convenient for you.

5.3. Viewing statistics

Statistics provide much useful information regarding the projects. You can see how much time the project will take according to given estimations, how much of the project has been done, how good your effort estimations have been, how much time has been spent on fixing bugs, etc. There are also some graphical statistics and employee related statistics.

5.4. Tracking changes

Tracking changes is a very useful feature if you're releasing minor updates rather swiftly. The change tracking system can show what changes have been made between two given dates (or Important dates specified in the project settings) and you can basically copy paste the returned result into your version history file. There are also other uses for the change tracking, such as determining which change could have created some unexpected behavior, etc.