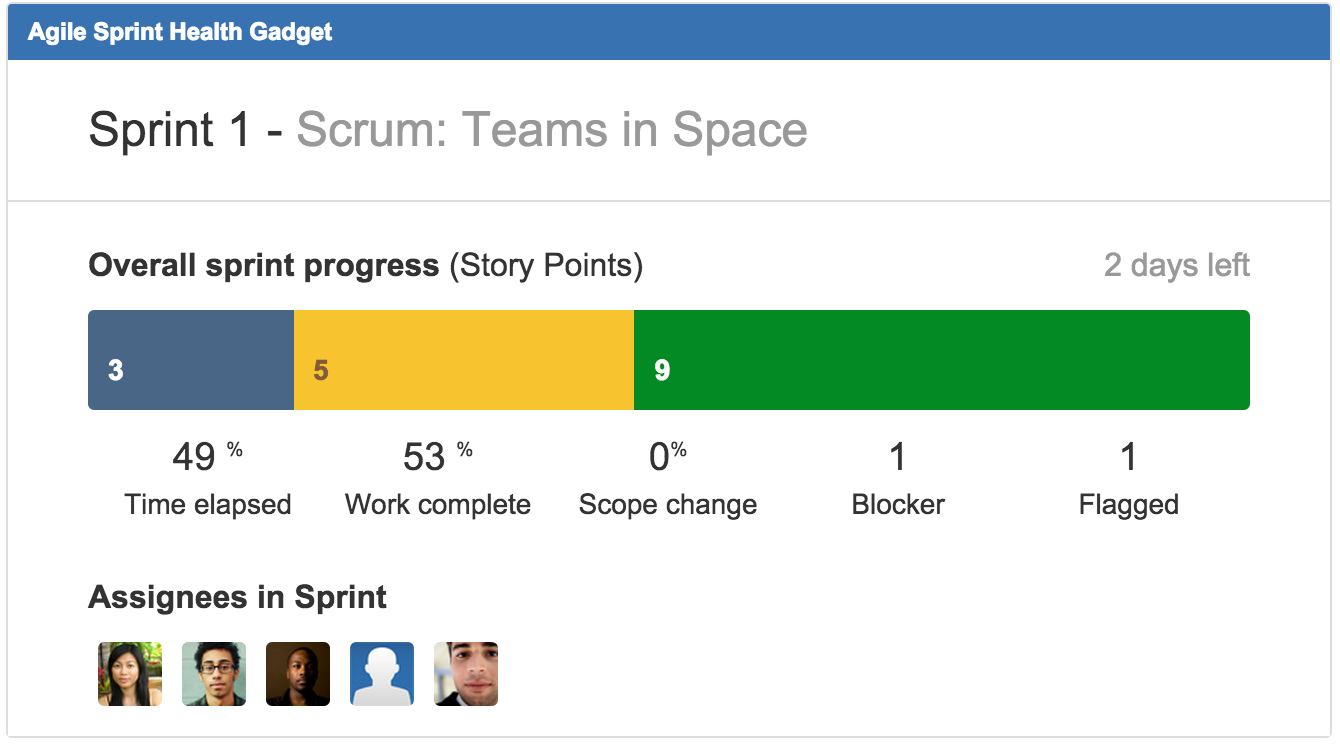
**Agile dashboard- Reports to time**

## 1: The Sprint Health gadget

The Sprint Health gadgetsummarizes the most important metrics in a sprint.



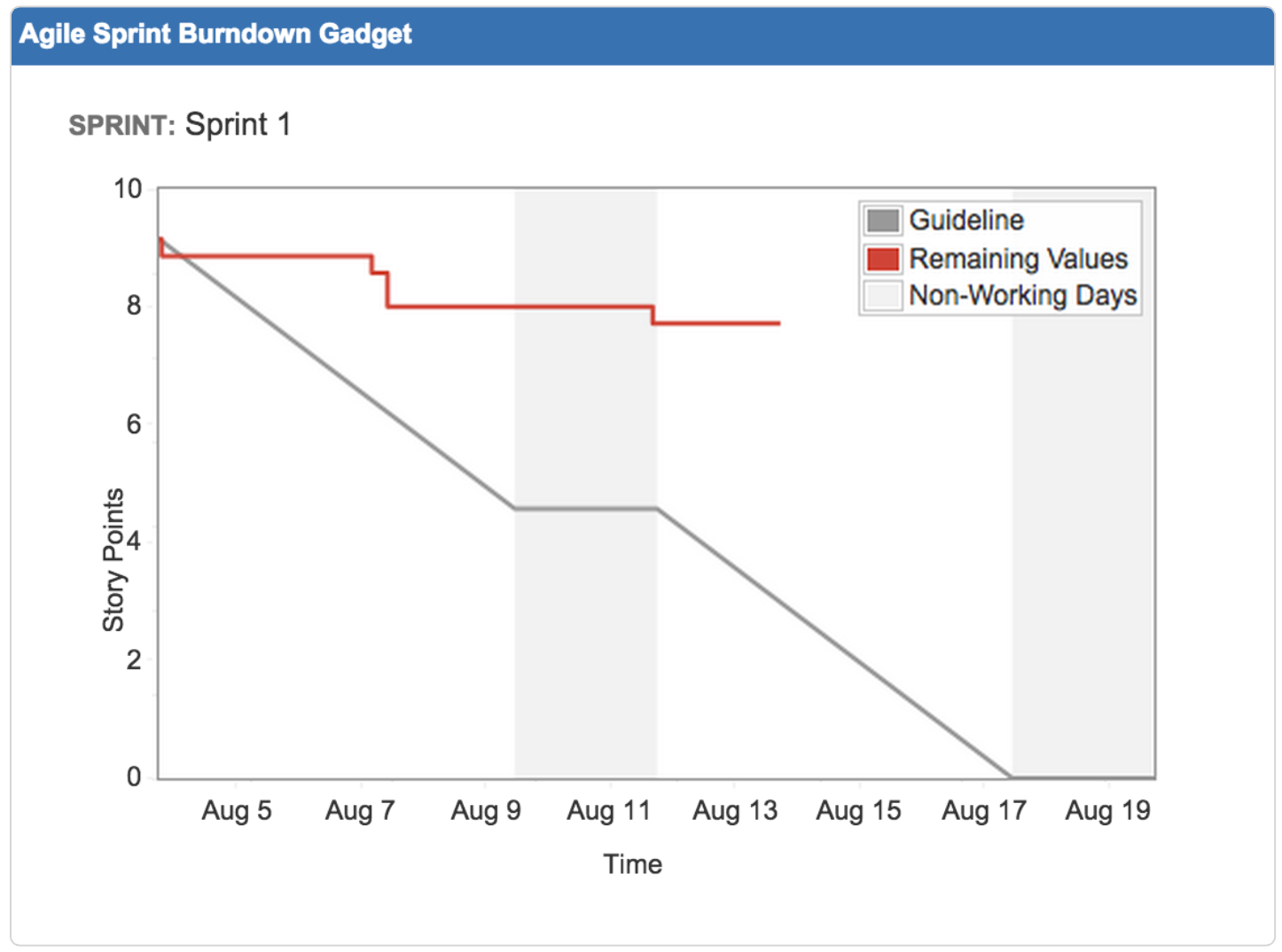
* Select the agile board you want to pull data from for the Board field
* Enable all three options underneath so assignees, sprint name, and board name are always visible
* Use the Next Sprint Due (auto) option for Sprint so you don’t have to update the gadget’s settings every sprint
* Hit the Save button, and bask in the glory of being super-informed

## 2: The Sprint Burndown gadget

The Sprint Burndown gadget shows your team’s track record for the current sprint. The beginning of the sprint is on the left and the end of the sprint is on the right. In general, we want to see the Remaining Values line stay under the Guideline, shown in grey.

For this chart to be accurate, estimation in the beginning of the sprint is critical. The chart uses the estimates to draw that guideline. What are the flat spots in the guideline? Jira Software has the ability to specify non-working days in your sprints.

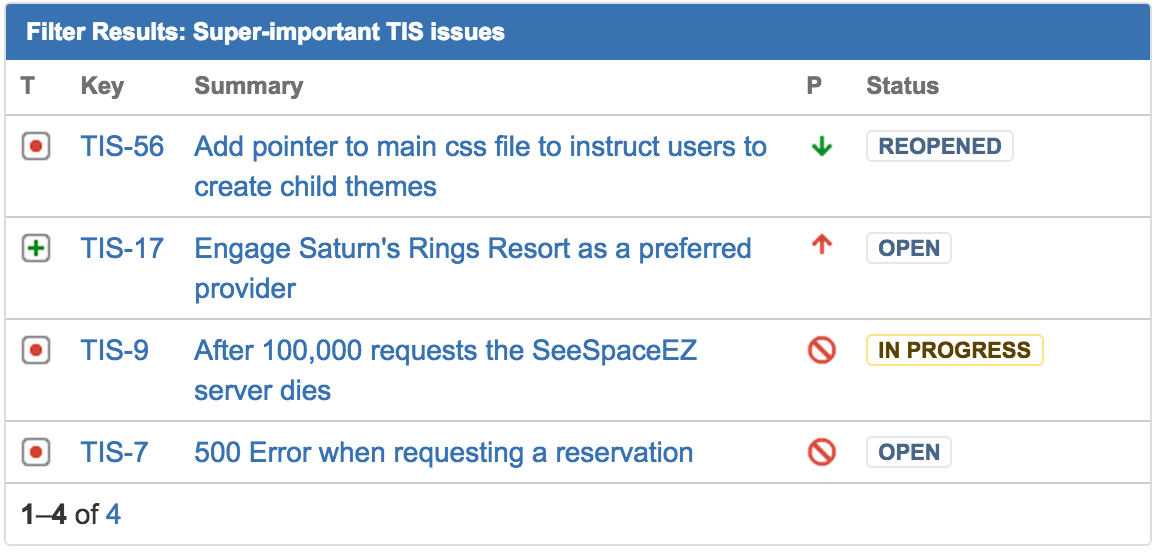
* Add the Agile Sprint Burndown gadget to your dashboard
* Select your agile board of choice for the Board field
* Use the Next Sprint Due (auto) option again



## 3: High-priority issues

I like to have a list of high-priority issues for my team in the upper right-hand corner. I can use a JQL statement to find any issue that has the Flagged field selected, as well as any issue that is considered a blocker. Why Flagged? Flagging an issue means that there’s an impediment for a team member to move forward. If an issue is marked priority = blocker, then that issue is globally important to the health of the project. Both are of concern to the team, but for different reasons. So your JQL statement will look something like:

* Add the Filter Results gadget to your dashboard
* Save the JQL above as a filter and enter its name in the Saved Filter field
* Adjust Number of Results as you see fit (hopefully your team doesn’t have more than 10 of these issues at a time!)
* Add Status under Columns to Display for easy reference



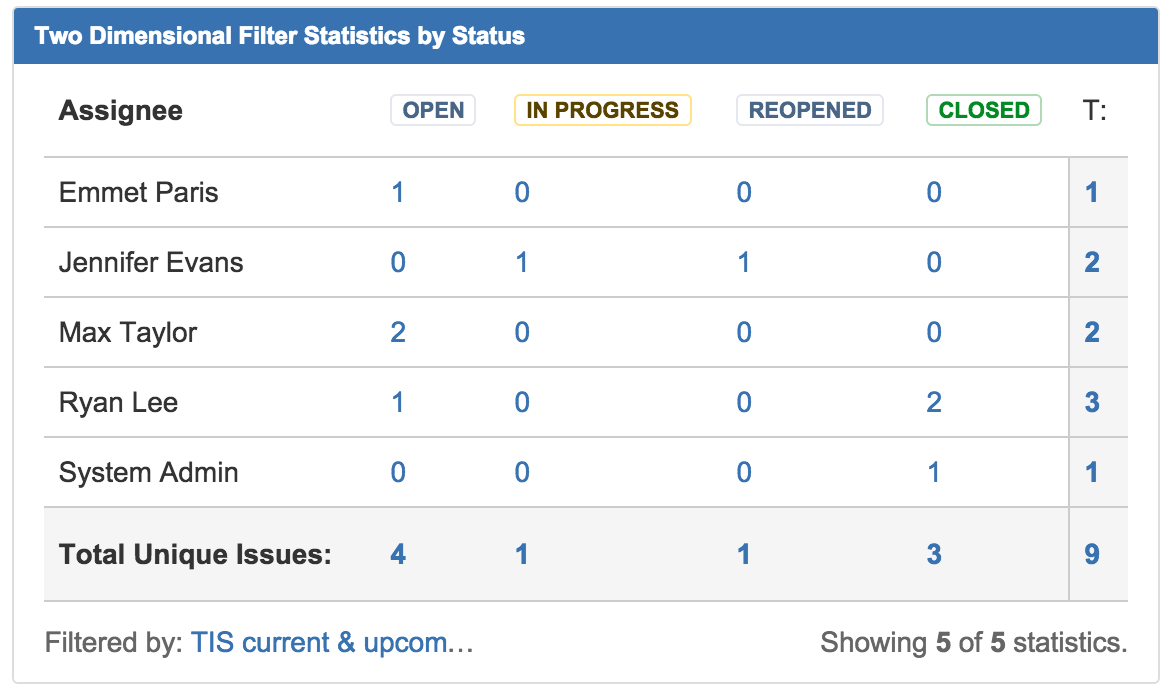
## 4: The Two-Dimensional Filter Statistics gadget

While a mouthful, the Two-Dimensional Filter Statistics gadget packs a real punch! You can configure it to slice a set of data so that it’s easy to find key areas of concern. Using this gadget, I want to know two things: how much does each team member have on their plate, and what are the relative priorities of work distributed throughout the team?

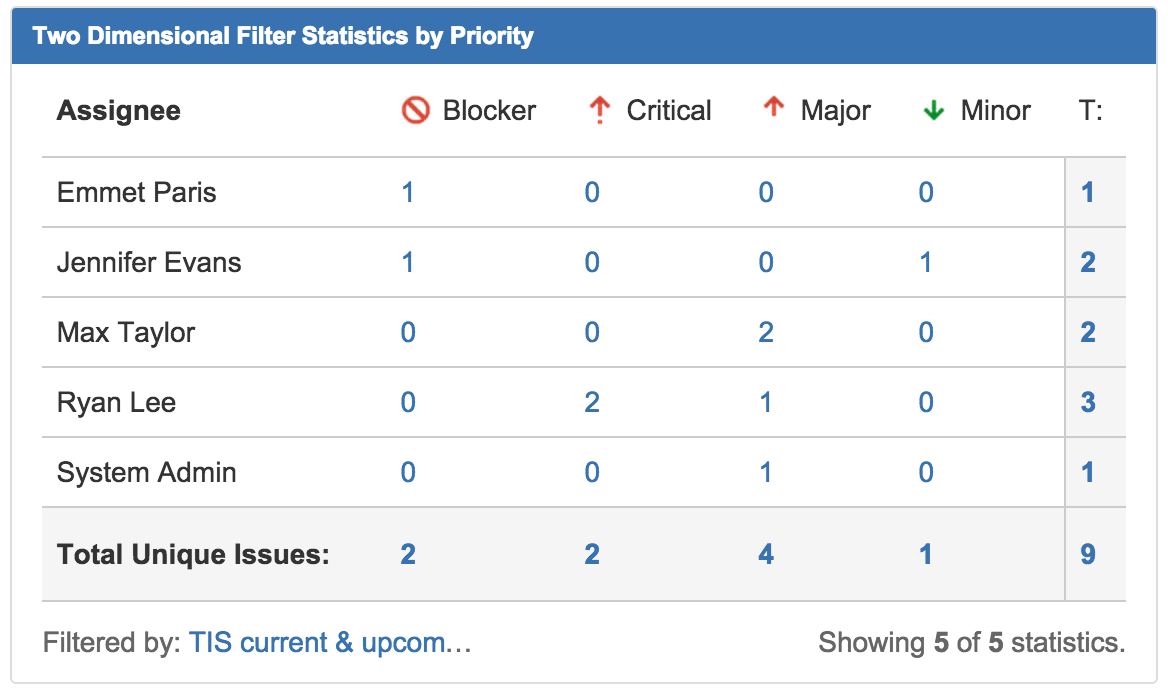
Let’s consider all of the open work in setting up our base filter. We can use the following JQL statement to show the work in the current sprint as well as future scheduled work:

Add the Two-Dimensional Filter Statistics gadget to your dashboard twice. We’ll configure the first one to show how much work each team member has on their plate:

* Save the JQL above as a filter and enter its name in the Saved Filter field
* Select Status for the X-axis field, and Assignee for the Y-axisfield
* Set Number of Results, to equal the number of members of your team
* Leave the rest of the fields set to their default values (or adjust as you see fit) and click the Save button



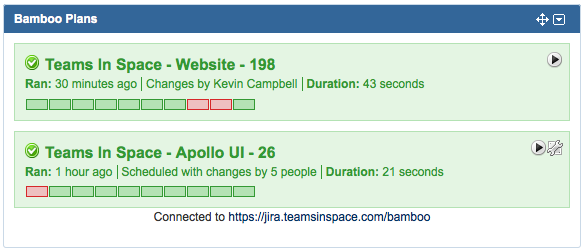
For the second gadget, follow the same steps as above, but select Priority for the X-axis field this time. Voila! In this example, we can see that one team member, Ryan, has 2 critical issues on his plate, while Max has no critical or blocker issues. Perhaps some issue shuffling and reassignment is in order.



## 5: Continuous integration

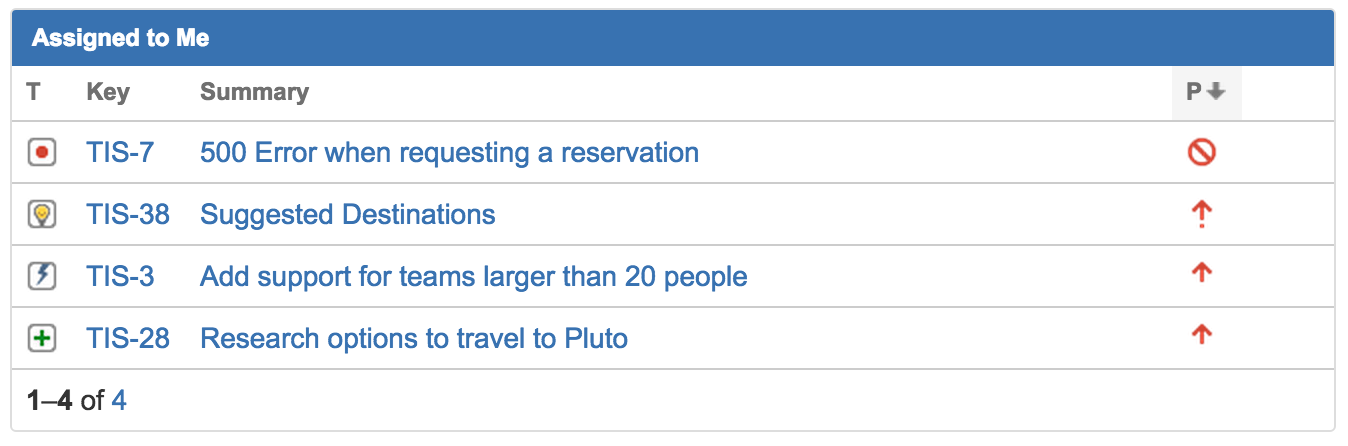
The most successful agile teams live and die by their continuous integration. I cannot overemphasize this! Consistent, effective investment in automated testing strategies by the entire team enable agility like no other effort. Using Atlassian Bamboo, teams can stay on top of build breakages, and ensure that automated tests are running in top condition. When the build breaks, everyone suffers. And when the build succeeds, everyone wins.

Ask your Jira admin to configure the Bamboo plugin for Jira to make the Bamboo Plans gadget available. A few other Bamboo and build-related gadgets come with it, too!



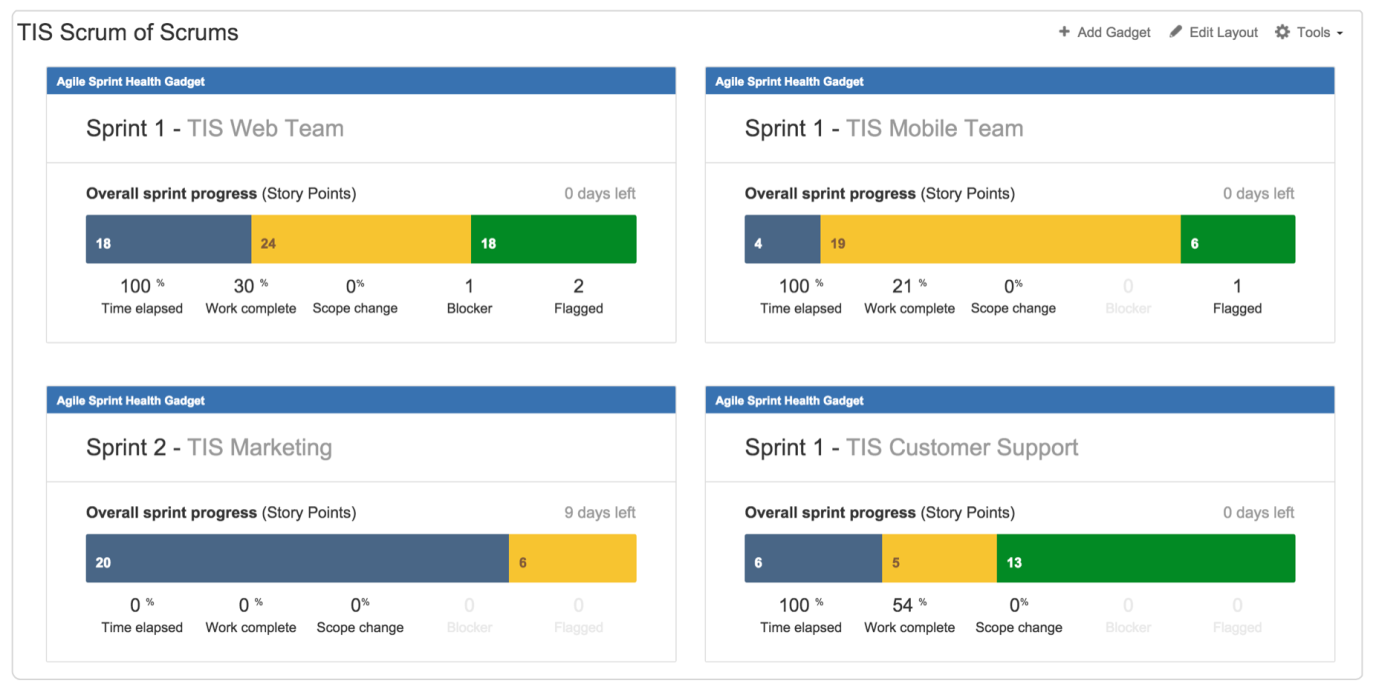
## 6: The Assigned to Me gadget

Some teams like to assign out work at the sprint planning meeting, and often times someone creating an issue will assign it to a specific team member right off the bat. You can use the Assigned to Me gadget so that each team member gets a gadget listing their work. Setting this gadget up is as easy as it gets. Just add it to your dashboard, leave all the default settings in place, and save it. Boom.



## 7: Level up

Teams don’t live alone. You can extend your dashboard to include data from multiple teams. Need to track multiple scrum teams together? Use several instances of the Sprint Health gadget alongside the Agile Sprint Burndown gadget to get a quick overview of how all teams are progressing towards a common goal.



|  |  |
| --- | --- |
| **Time Tracking Report \*** | Shows time tracking information on issues for a particular version of a project.  The table in the report shows the issues within the version:   * There are four time tracking fields as follows:   + **Original Estimate** - The original estimate of the total amount of time it would take to complete this issue.   + **Estimated Time Remaining** - The current estimate of the remaining amount of time it would take to complete this issue.   + **Time Spent** - The amount of time spent on the issue. This is the aggregate amount of time that has been logged against this issue.   + **Accuracy**- The accuracy of the original estimate compared to the current estimate for the issue. It is the difference between the sum of the **Time Spent** and **Estimated Time Remaining** fields, and the **Original Estimate** field. * If sub-tasks are enabled, the \*'∑'\*column at the right of the field shows the aggregate time tracking information for each 'parent' issue (i.e. the sum of the issue's own values, plus those of its sub-tasks). * The last line of the table shows the aggregate time tracking information for the whole version.   The report also includes two bar-graphs (above the table), which represent the aggregate time tracking information for the version:   * The first bar-graph **('Progress')** shows the percentage of completed issues (green) and incomplete issues (orange) in this version: https://confluence.atlassian.com/jiracoreserver073/files/861257295/861257300/1/1481516808892/timetracking_report-completionbar.gif * The second bar-graph **('Accuracy'** -blue) shows the accuracy of the original estimates.   The length of the **Accuracy** bar compared to the **Progress** bar indicates whether the issues in this version are ahead of or behind schedule. There are three cases:   1. *The issues are on schedule with the original estimate.* The **Accuracy** bar is completely blue and is the same length as the **Progress** bar above it. https://confluence.atlassian.com/jiracoreserver073/files/861257295/861257299/1/1481516808841/timetracking_report-accuracyonschedule.gif 2. *The issues are behind the original estimate (i.e. will take longer than originally estimated).* The **Progress** graph is longer than the **Accuracy**graph. The blue region represents the original estimated time, and the light-grey region is the amount of time by which issues are behind.  https://confluence.atlassian.com/jiracoreserver073/files/861257295/861257298/1/1481516808783/timetracking_report-accuracybehind.gif 3. *The issues are ahead of the original estimate (i.e. will take less time than originally estimated).* The **Accuracy** graph is longer than the **Progress**graph. The blue bar represents the original estimated time, and the light-grey region represents the amount of time by which the original estimates were overestimated. https://confluence.atlassian.com/jiracoreserver073/files/861257295/861257297/1/1481516808741/timetracking_report-accuracyahead.gif   When generating the time tracking report, consider the following settings:   1. For **fix version**, choose the version on which you wish to report. The report will include all issues that belong to this version, that is, all issues whose **'Fix Version'** is this version. 2. For **sorting**, choose how the issues in the report will be sorted:    * **Least completed issues first** — shows issues with the highest **Estimated Time Remaining** first    * **Most completed issues first** — shows issues with the lowest **Estimated Time Remaining** first 3. For **issues**, choose which issues will be included in the report:    * **All** — includes all issues assigned to this version    * **Incomplete issues only** — excludes issues which are either completed (i.e. have an **Estimated Time Remaining** of zero), or are not time-tracked (i.e. do not have an **Original Estimate**). Note that issue status does not affect which issues are displayed. 4. For **sub-task inclusion** *(note: this will only appear if* *sub-tasks* *are enabled)*, choose which sub-tasks will be included in the report, for all parent issues that belong to this version:    * **Only include sub-tasks with the selected version** — includes an issue's sub-tasks only if the sub-tasks belong to the same version as the issue    * **Also include sub-tasks without a version set** — includes an issue's sub-tasks if the sub-tasks belong to either the same version as the issue or to no version    * **Include all sub-tasks** — includes all of an issue's sub-tasks, regardless of whether the sub-tasks belong to the same version, some other version or no version. Note that sub-tasks which belong to this version, but whose parent issues do *not* belong to this version, will always be included in the report. |