Zephyr for JIRA Brings Process, Visibility and Accountability to the Release Schedule

CASE STUDY

Q: Can you tell us a little about ProCare?

Sure, ProCare provides transportation and translation services for the workers compensation industry. We work with clients throughout the United States and have developed a cutting edge scheduling system that provides both us and our clients with up to date information regarding the status of their appointments, claims and records. Our transportation service includes reliable, clean and climate controlled vehicles. We have multi-lingual drivers who provide extraordinary customer service, always willing to meet any special accommodation needs.

Our translation services include; interpreters, translation and transcription services in over 200 languages. Our staff specializes in medical, legal and technical documentation that is used in the workers compensation industry.

Q: How does software play a role in your company?

When ProCare launched the founder could not find a commercial software solution to meet their needs. We then decided to build PRIME, our core business application from scratch. Over the past 5 years we enhanced PRIME to include all of the functionality that our employees need to be able to take orders, schedule and track appointments, negotiate with providers and report to clients. The project keeps several developers and some internal QA staff busy full time with 3 to 5 concurrent projects.

Tampa-based transportation and language services company, ProCare, uses Zephyr for JIRA to manage software testing. The solution brings process, visibility and accountability to the release schedule.

Today, I had the pleasure of speaking with Jim Begley who manages ProCare’s software development projects. He was kind enough to share some of the testing challenges his team faced and how Zephyr for JIRA allows his project team to incorporate testing into their JIRA workflows.

Headquarters: Tampa, Florida
Employees: 110
Founded: 2007
Products: JIRA, Confluence, Zephyr for JIRA
Website: www.theprocare.com
Before Zephyr for JIRA, testing was done ad hoc using a combination of JIRA, Confluence and spreadsheets. It was very manual, time consuming and we would find bugs months after the release. We needed a more rigid and structured approach, but not something with process overkill. Releasing dates relied heavily on gut instinct. Zephyr for JIRA gives us hard data and accountability to determine what is ready for release.

There was a huge improvement in how we test, it is much more structured and we now have a repeatable process in place. We no longer have debates over releasing; now we have hard data to support our decision making. The application is extremely easy to use; we can create and execute test cycles, track progress, file new bugs or add existing ones, and easily prioritize what to test vs. what to differ. I definitely like the speed in which you are adding functionality as you added some of my requests in almost each new release.

There are a handful of metrics that we use and Zephyr for JIRA makes it possible to connect test cases to each of these bug metrics inside JIRA.

- number of new bugs compared to the previous release
- the severity of bugs compared to previous release
- percentage of bugs found in-house
- percentage of bugs reported by the client
- percentage of hours used to fix bugs in relation to the development effort
- time to close

We have seen significant increases in productivity now that our process is repeatable. We use historical information for planning and reuse test cases so that we do not have to start each new project from scratch. Instead of cherry picking tests, we prioritize what to test and what not to test.

I cannot put hard numbers on it, but we definitely see time savings from the process improvements and reusability. We have more accountability and visibility into what failed vs. what passed. This allows us to write more granular test cases, reducing the total amount of bugs as well as the severity of the bugs that get released. Overall, this improves the quality of our software. Also there is no training needed since it looks and works like JIRA, everyone was able to pick it up quickly.

There are several test management solutions out there, why did you choose Zephyr for JIRA?

We manage everything in JIRA; requirements, bugs, time sheets and project management. When we learned that there was an application that would allow us to do testing inside JIRA, we decided to try the free version. We attended a webinar, watched the videos and read the online documentation. Zephyr’s approach really resonated with the team. After playing around with the tool and evaluating other options, this looked like the best product for our needs.

Zephyr for JIRA gives visibility into testing and bug metrics. Before we only had bug metrics, but now we have test metrics that allow us to see whether a bug is really one bug or three smaller bugs made up into one bug. We can easily see the top defects that are impacting testing and use burndown charts to predict when the testing cycle will be completed. The ability to easily navigate between test cases and link them to issues gives us complete traceability all within the JIRA UI.

How is Zephyr for JIRA working out for you?

There is a huge improvement in how we test, it is much more structured and we now have a repeatable process in place. We no longer have debates over releasing; now we have hard data to support our decision making. The application is extremely easy to use; we can create and execute test cycles, track progress, file new bugs or add existing ones, and easily prioritize what to test vs. what to differ. I definitely like the speed in which you are adding functionality as you added some of my requests in almost each new release.

What were some of the challenges that your team faced prior to Zephyr for JIRA?

Q: How do you quantify each of these benefits?

I cannot put hard numbers on it, but we definitely see time savings from the process improvements and reusability. We have more accountability and visibility into what failed vs. what passed. This allows us to write more granular test cases, reducing the total amount of bugs as well as the severity of the bugs that get released. Overall, this improves the quality of our software. Also there is no training needed since it looks and works like JIRA, everyone was able to pick it up quickly.

Q: How do you measure or assess the quality of your own products?

There are a handful of metrics that we use and Zephyr for JIRA makes it possible to connect test cases to each of these bug metrics inside JIRA.

- number of new bugs compared to the previous release
- the severity of bugs compared to previous release
- percentage of bugs found in-house
- percentage of bugs reported by the client
- percentage of hours used to fix bugs in relation to the development effort
- time to close

Q: What specific benefits did you receive from using Zephyr for JIRA?

Zephyr for JIRA gives us visibility into testing and bug metrics. Before we only had bug metrics, but now we have test metrics that allow us to see whether a bug is really one bug or three smaller bugs made up into one bug. We can easily see the top defects that are impacting testing and use burndown charts to predict when the testing cycle will be completed. The ability to easily navigate between test cases and link them to issues gives us complete traceability all within the JIRA UI.

We have seen significant increases in productivity now that our process is repeatable. We use historical information for planning and reuse test cases so that we do not have to start each new project from scratch. Instead of cherry picking tests, we prioritize what to test and what not to test.

Q: How do you quantify each of these benefits?

I cannot put hard numbers on it, but we definitely see time savings from the process improvements and reusability. We have more accountability and visibility into what failed vs. what passed. This allows us to write more granular test cases, reducing the total amount of bugs as well as the severity of the bugs that get released. Overall, this improves the quality of our software. Also there is no training needed since it looks and works like JIRA, everyone was able to pick it up quickly.

Q: How do you measure or assess the quality of your own products?

There are a handful of metrics that we use and Zephyr for JIRA makes it possible to connect test cases to each of these bug metrics inside JIRA.

- number of new bugs compared to the previous release
- the severity of bugs compared to previous release
- percentage of bugs found in-house
- percentage of bugs reported by the client
- percentage of hours used to fix bugs in relation to the development effort
- time to close