**Please elaborate on your experience with Atlassian JIRA, Confluence, Bamboo and any other Atlassian products (Please go into detail and talk about your experience with JIRA in terms of admin)**

I am a Devops Engineer having an extensive knowledge on JIRA and Confluence. My typical responsibilities include creating users in user management, setting up the projects list and troubleshooting the issues. I will be helping up the client in setting up the dashboards which will help them in tracking their status in graphical representation. We have JQL tricks plugin through which we can filter the issue with issue links and issues assigned to inactive users (user left company is inactive). Worked on labelling and Linking issue in JIRA which helps us to recognize an issue in more flexible way and linking issue helps in associating the two issues on the single or multiple JIRA servers.

In our company, Confluence work as a wiki, as we document all the stuff. In which everyone can login and share their ideas. Knowledge on the Macros, it is the add-on separately installed component that modify the confluence.

**Elaborate on your experience with Subversion plus any other source code management systems (CVS, GIT, etc)**

Coming to my experience on GIT, In Cisco I have installed GIT on Linux. As GIT allows us to track revision and changes in files and it is typically used for software development. IT is more intended to be used in distributed mode which means, every developer checking out code from central repository/server will have their own cloned repository installed on their machine.

Let’s say if you are stuck somewhere where you don’t have network connectivity, like inside the flight, basement, elevator etc. You will still be able to commit files, look at revision history, create branches etc. This may sound trivial for lot of people but, it is a big deal when you often bump into no-network scenario. And also, the distributed mode of operation is a biggest blessing for open-source software development community. Instead of creating patches & sending it through emails, you can create a branch & send a pull request to the project team. It will help the code stay streamlined without getting lost in transport. GitHub.com is an awesome working example of that.

CVS server execute certain scripts. I have written the script that works is to validate the CVS commits a comment used by the developer. The process involves setting up the CVSROOT and verifymsg file to run a script, when a code is checked-in.

**Elaborate on your experience with continuous integration and version control systems**

I have worked for four years on CI and VC.

We prefer Jenkins, as it is an open source tool with plugin built for CI purpose. It helps to keep a track of version control system to intimate and monitor a build system if there are any changes. There are many uses with GIT (VC) as it automates the backups and sharing on multiple servers and branching.

**Elaborate on your experience with optimizing the continuous integration and build systems.**

Jenkins is an open source tool with plugin

It is critically important for every team to get their development workflow in order. Two best practices we follow is to have the Continuous Integration and continuous deployments. CI is the practice of validating each change done to the code base automatically and as early as possible. In the company, we follow maintaining a code repository and automate the build process. Make the build as self-testing. Every commit should be built. And we keep the build fast. We will test in a clone of the prod environment. And we will automate the deployment.

**Elaborate on your experience applying customization and third party add-ons to extend JIRA functionality**.

Zephyr is a third-party plugin for JIRA. Zephyr is ready to install tests. JQL tricks plugin.

Eazybi is a powerful reports, charts and dashboards add-on for JIRA.

It's an easy drag and drop tool for visualizing and analyzing JIRA Issue Data.

helps to analyze JIRA Issues by Standard and Custom issue fields.

Identify Trends and Top/Bottom Performers.

Starts from Summary Overview and Drill deep in to details.

create Pivot Table Reports and visualize data with many Chart Types and Publish eazyBI reports as gadgets on JIRA Dashboards.

Import additional data from: CSV or Excel file

SQL & REST APT data sources and combines them in reports with JIRA.

No issues created/resolved or estimate/ actual hours and time spent in workflow transitions using bar, char and pi diagrams.

Example: Generating a report Issues due

Issue created, Issue resolved, Issue closed, Issue with date, original estimated hours, remaining estimated hours, Hours spent, Sub-task created, Sub-task due, Sub-task resolved, Sub-tasks closed, Transitions to status etc.

Now let's create a report for Issues Due in different projects.

Basically, the chart is divided in to 3 parts.

1. Page

2. Rows

3. Columns

In this scenario as we are looking in to Reports Due, page is Projects i.e., the projects that are currently in progress

Rows: Priority of the Issue and Assignee

Columns: Add Issue Due and status in the columns

We can also use conditional cell formatting tools to filter the results.

Also, we can publish the report as a gadget to JIRA Dashboard page.

**Elaborate on your experience with Scons (additional knowledge of other make systems is desirable)**

Scons build scripts that do everything from generating code for language bindings to building libraries. In my experience with this larger project has also convinced me to move over my JIRA tracker because of all the other things I should be doing, the most important thing is moving over the build system for no particular reason at all.

**Elaborate on your experience managing JIRA permissions, including project-wide, issue, space, and repository access**

Once we had an issue , where the user when clicks Tests > Search Test Executions we get the 104 error:

I have tracked the issue to the Ad Hoc test cycle in the Unscheduled Version in Zephyr. If we look at that cycle, it shows as if it has test cases are in it but when we open it none are there. Unfortunately, the Unscheduled Version is not one that we can delete because it is the system generated version that is created when we first start using the Zephyr add in.

We need to get this fixed because searching test executions is critical for UAT and very useful for monitoring the NuWare activity that is going on in our company. Then I checked the below permissions and tested and the 104 error is fixed:

1. Below Addon User Exists:

=> Navigate to Administration —> User Management and check for the user "addon\_com.thed.zephyr.cloud"

2. Group membership for Addon User: The User addon\_com.thed.zephyr.cloud should be a member of all the below groups.

=> atlassian-addons or atlassian-addons-admin

=> jira-users

3. Global Permission: The group "atlassian-addons-admin" has JIRA Administrators permission

=> Navigate to Administration -> System -> Security -> Global Permissions

=> Under permissions JIRA Administrators contains atlassian-addons-admin as a group

4. Project Role assigned for the Addon user "addon\_com.thed.zephyr.cloud":

=> Navigate to Administration —> System -> Security -> Roles

=> Check the role atlassian-addons-project-access Cantains the User - addon\_com.thed.zephyr.cloud as a member

5. Project Permission: Add the role to all permissions

=> Navigate to Administration -> Projects -> <Select any Project> -> Administration -> Permissions

=> Add the role "atlassian-addons-project-access" to all project Permissions

All the permissions are in place again. Please reindex both JIRA and Zephyr for JIRA (Project & Executions) respectively by navigating to add-ons> under zephyr for jira select "General Configuration">re-index.