**How do you document your test cases?**

1. XRAY test cases
2. Test cases are **stored on Google doc** and also put doc in User Story in Jira.
3. **Excel** online so everyone can reach
4. **ALM**
5. **Don’t document-** straight I automate
6. Using **excel files** and **adding** test cases **under the user story** on the Rally board
7. **manual test cases are** in word format attached to Jira, **automation in Cucumber feature file**
8. **Under ticket comments**, for active sprint tickets. We have an application Teslink, it is not part of Jira, keeping regression test manual steps there
9. In Jira, for each US creating a Test Ticket and linking to US, tracking my own Test Ticket along with US itself
10. **No test cases**
11. Test cases are jira
12. No test case cause BA write test case under the US
13. Jira
14. In jira

When you find a bug how do you handle it?

1. Once I found a bug I report it to PO and Qa manager.
2. **I confirm it with my peer and make sure it is not duplicated.** If I am not sure about the requirement, try to clarify it with BA and the respective developer. Then create the ticket.
3. The first thing I will retest is to make sure it’s a legit bug and then follow up with my developer and retest again after the dev checks,  **it’s a small issue I don’t report,** but if it’s a major one I will report in Jira and follow bug fixing steps.
4. We document them in the Jira as Bug. Then providing steps to reproduce these issues and assign them to the corresponding developer. We do have a QMetry plugin where we store all Test Cases and Bugs related to it. (as a reference)
5. If I find it during the dev work, just create test case under the user story in Jira (Rally for my case) and share it with the developer. Dev fixes and pushed the changes and test again. if passes, move the user story to completed and demo to PO. if the defect is found after PO accepts the user story, I create a defect story in the Rally.
6. **I do report to a dev** through Microsoft teams. **if they can’t handle it within sprint** **they ask for open bug tickets in Jira.**
7. Jira. The Dev team makes a decision whether to fix for the release branch or develop.
8. Double checked with my colleague, then chat with dev. For active sprint tickets: If he agrees as well, then I am creating a slug(type of bug sub-task under related ticket (user story or bug).
9. I create a bug ticket in backlog, if urgent track into current sprint board after discussing with PO
10. Reporting in Jira, prior checking with the team and giving heads up that will be opening a bug ticket
11. **we don’t report it, we just tell to developer**
12. **First discuss with other teammates in chat or email,** if it is a vague issue. After making sure:

* If it is a **sprint ticket-related bug,** I assign the ticket back.
* If it is a **regression bug,** I open a ticket.
* If it is a **non-functional issue,** or a user experience issue, or a requirement gap, I **talk to business people** or architects and ask for further opinion (it may lead to a new story)

1. Contact the team via Slack. Ask for expectations, share the screenshots/data, etc. Then make a call with the dev if needed. **Ask for clarification to the PO if needed.** Then create a bug ticket with all the detailed info in Jira. **If urgent move it to the current sprint, if not leave it at the backlog.**
2. First re-test, then make sure conditions are valid (correct version, correct browser, etc), reproduce again, **file a bug report in Jira**. Needs to include steps to reproduce, conditions, version, and evidence. Make sure to follow the bug process. Whenever resolution changes to “fix” then re-test to validate the fix and close the bug if satisfied.
3. Track them on project management tool, report with evidence steps to reproduce and severity.

How do you solve conflict with your developer?

1. **We are in the same team and working for the same purpose.. Communication is the key** and testers should involve the development process by asking developers if they are building the right product. So that they will be less resilient against their mistakes. Being aware that "I may be wrong" is also important. **If we are %100 sure, then we need to ask him to go and clarify the issue with BA or PO.**
2. Prevent them, make sure you are on the same page before jumping to conclusions.
3. Conflict is a big word. Usually, we have different views on bugs or how functionality is supposed to work. What I usually do is set up some time for us to talk about the feature/bug in question so **we can both explain our views.** Based on requirements and end-use cases we should be able to figure out what is the right way to develop and test the feature/bug. **If we are still in disagreement we involve scrum master/PO** since they are in charge of requirements and can provide the right direction.
4. Communication is the keyword to solve any issue. **Listen carefully, explain detailed**. Everyone has a different point of view.
5. I had more conflicts with project people than developers. It is usually the time crunch. I ask for extra resources to accelerate testing or ask for an extension or ask for risk analysis as a team (open a defect in case some part of it is failing)
6. **scheduling a meeting including tech lead**
7. Just a **professional discussion**, showing the requirements and pointing out the mismatch
8. Discussing with them if it is a bug if so in the scope of the ticket, if not agreed to clarify with PO
9. Talking them, given a demo, if we do not agree on the acceptance criteria, then s**etting extra 3-amigo-meeting** with PO, to solve the issue
10. We simply talk. Back and forth. If cannot agree on expected behavior, go together to the PO.
11. **i don’t have any conflicts within 2 years**, but we had topics we did not agree on. in that case BA and PO are the best people to talk to.
12. Talking it out and respecting their opinions
13. the key is always **close coordination,** and a good relationship with them solves most of the problems.. if the conflict is based on dev work, then I share what I know/understand from the requirement documentations(can be user story, API doc, schemas, or architectural docs) and try to find out who’s missing what. if the conflict is personal, that's no good.. I go to my manager and mostly update him for possible issues even it comes up.
14. 1. Try to resolve issues between each other (Listening and trying to understand each other’s opinion and point of view) 2. Let the Test and Technical Dev Lead know, and add them to the chat or meeting. (From the higher level it can be easily resolved; where they will give some clarity)
15. **I try don’t have any conflicts**, but in the worst scenario communication is the key.
16. No conflicts.