Think Aloud Protocol

Anonymous

January 13, 2021

Table 1 (P1 ... P6)

- TASK: OpenStack Events P3
  - Events and activities within the community (*Look up the location and dates of the next PTG and share their response to the IRC channel.*)

    OB1 Can you walk through the exercise and explain each possible step verbally? That is, your though process.
    - OB1: can you think aloud? Explain your steps and what the functions do.

P3 : Now, I am searching the keystone documentation to see which functionality I am familiar with within the most recent release of OpenStack. I have Identified the first, functionality: A function that creates an OpenStack user with federated identity. Now, I am reading what the code does, ... it takes four-parameter and returns a dictionary containing the user reference. Next, I am searching the keystone documentation again, ..., I have found another functionality that returns a consumer and the consumer's secret. [The search continues...] In the manager class, I have seen a function that dynamically calls the backend. This function serves as the default pivot point for authenticating backends, [and the search continues ...] Last, I have found a function that validates authentication from a query string. I will stop the search here because I fill satisfied with this task.

- OB1 to P6
  - TASK: OpenStack CI/CD P6

I will like you to think aloud in this task if you don't mind . . . P6: No problem that is on by me. So, go ahead. P6:

> I am opening a new tab on my Firefox browser and search for the URL that mentors provided, So from what the mentor M5 said, OpenStack uses ZUUL, so I have to keep the documentation open at all time to keep track of what I want to do. [Hmm I really like the way mentors have arranged the documentation links everything work so nicely.] Now I am looking at the ZUUL page and reading the instructions [. . .] OK concerning the checks and rechecks it seems I am a bit lost here, let me figure out the way out [Fidgeting with fingers . . .] OK I figured it out, the key point here are What to do if a test job fails? And What is elastic-recheck. Let me see if I can understand these charts . . . [oh yea we go] It show cases of both failure and success using different colors for both. In this particular case, most of the CI jobs failed. Let me read the failure message to know why those jobs failed. [silence] OB1 what are you thinking about the failed jobs? Ah yea, I just want to make sure I understand how the checks work. The system looks too complex, I really wander if I could ever understand all this with the help from mentors [Hmm] I am looking further down to see the other error messages in red. NIce! I found the link and the bugs, let me just copy paste the link to the IRC channel and there we go. I am done with the exercise.

Table 2 (P7 . . . P12)

- observation :P7: and :P10:

Testing Framework exercise

  - OB1 to :P7: when you are ready to start the exercise, can you say aloud what actions you are doing at each step?

    O.K I am looking through the test suits and I want to run several unit test cases and

    an integration test. First, I am using the tox framework to run unit testing, so I call the tox command on my terminal [typing . . .]

    Next I am specifying the parameters to tell tox that I want to run a unit test [typing. . .] [Hmmm, did I write the command correctly? Let me verify from the docs]

I want to double check from the docs, I am searching for the section that specify how to run testing with tox,... OH here we go! I found it and I made an error in the command.

Let me correct it. [I have realized that the life of a developer is constantly checking the books ... Smiling] O.K so not I press the return key and see what the output looks like

... stip stip stipOK everything looks good from the result. Next, I am running the second command, ... looks good, third, I want to test individual modules to see how the tox module penetrate systems

... on Done, now, let me just show the mentor to be double sure I did everything well. [call the help from a mentor] Please, look if I did everything as expected

M3: "Yes, looks perfect." Ah! so I am ready for the next exercise.

- OB1 to P10: As you get ready to run the integration test, I will like you to verbally express your though process. Is that O.K with you? .. Sure! OK let's go when you are ready.

  So, I am using the a different project; Neutron as the project that I selected. I am moving into the folder by typing cd Neutron ENTER. Next I have a habit to always make sure the repo I am using is git, I type git status, and git pull [to get the latest version from remote]... at this point I am asking a mentor to come and break the code as the. [signals a mentor next on the table.] OK let me give way for the mentor to make some changes [PAUSE for 2 min] OK let's continue... So, now I am typing the test command with option to running an integration test, I am following the instructions in the mentors' manual. [Test is running and displaying some output messages, I am scanning through the messages to follow up what is happening ...] Hmmm Error messages! Let me see what went wrong in the code that the mentor break... I am reading the first line on the Error message... OK, it point to line 967, let me see the details below. [OK] it's becoming clearer now what went wrong, I see that the code want to read a dictionary's values but can't find the dictionary and secondly, there is an array index out of range Error, So I am going inside the code now to fix the problem since I know which file and line number to access. I am using my vim editor passing the filename and plus the line number to go directly to

where the error was identified, now, I see that a block of code was commented out, and the names of both the dictionary and array matches those in the error messages, I am uncommenting the code block, ... saving the file and exit the vim editor. Then, I am re-running the command to see it fixes the problem.... [Eh Viola! It works] the test results shows 100% successful

Table 3 (P13 ... P18)

OB1 Could you describe the task as you are performing it?

OB1 observing :P15:

This particular task is about searching for documentation online and extracting useful information that we will use later in the training. So, I am opening a new tab on my Chrome browser ... typing the search query and looking through the search result. There are tons of documentation all over the place that makes it had for me to smoothly follow what is happening in the community {OB1: Are you confused on how to proceed?]

Not really, because as I follow the instructions that mentors gave, I think I am able to quickly access documentation ... it's a lot easier than I though because I think mentors have classified and organized the docs. I can now stop worrying about searching the right doc and focus only on the technical details. This too I fond help to increase my productivity. [OB1] So, what is your next move on this task?

- TASK: Write and share your complete commit message with someone sitting next to you. Also, give them feedback on their commit messages.
- **OB1** observing :P18:

  At this point I am adding the changes to the stage area ... with git add filename, then I am committing the changes

  with git commit -s -m ENTER to pop up the editor, which in my setting is Vim.

  I am writing the message following the best practices that mentors taught us today.

First, for my title message ... Purposing a new feature for Neutron backend.

Then I am skipping two lines to write the body of the message. The body will

have three paragraphs, 1. I am explaining what problem I want to solve here with this new feature.

... My intuition is that Neutron should be able to hide all internal networks from outsider. [OB1]

Are you certain that such functionality is not yet implemented? [Emmm]

I don't know for sure if implemented alredy or not, but, I am only

trying to learn how to do the contribution workflow. There is no harm in trying, I guess.

In this second body paragraph, I am explaining that Neutron will have a function to hide VMs visibility.

Last, I am explaining how I implemented the new feature [...] typing. Then, I an pushing the commit upstream.

.... [Silence]

OB1: are you thinking bout something?

Not really, I am notifying my teammates to look for the commit that I just pushed upstream

and to review it.

Next, I am accessing my teammates commits to see the one that I will like to review and provide feedback.

I have chosen P13's commit to review.

I am in the Gerrit review system now, I am looking at P13's username to make sure it matches the system .

I can see the change request now.

Looking the commit title, ... [OK] Looks good, and explicit.

two skipped to line ... good, body message three paragraphs, excellent. Now looking indetail the body.

paragraph-1 . . . .. Ok explaining the problem before the fixes .. looks good
paragraph-2 . . . explaining what he is proposing . . . good
paragraph-3 . . . explaining the result of the changes, the scope of future improvement, excellent.

Writing my feedback: Your commit message is commplete and explicit. you respected the best practice and provided details.
I am approving your request with a +2

Table 4 (P19 . . . P24)

OB1 As you carry out this task can you say aloud what you are thinking?

observation :P23:

[– Oh here we go] I can imagine we have covered this much materials. With the help from mentors, I have been able to navigate to over four different projects and making changes, something I could not have done on my own and without the opportunity of joining this OUI training. So far, I have been able to made changes to the following projects: Keystone, Neutron, Swift and Sahara. [pretty challenging . . .] I also noticed that each of these projects have their unique objective.

-TASK: How to modify a patch within a chain, use Gerrit to explain. You can think aloud when you are doing the task.

- :P24:

I will need many '^' equally as the patch that I am editing from the top of the chain.
So I am doing the git restack command [branchname] then git will
figure out the appropriate git rebase command in the given context.

I will go now to Gerrit and show you have I can do this

I can edit the patch directly here or only the commit message.
I can also modifying the author(s) of the patch.
Gerrit is not getting interesting to work on.

Table 5 (P25 . . . P30)

TASK Getting to know your project

observation

:P26: I want to look for the next PTG, 2019 on the OpenStack event page.
I am opening a new tab on my Chrome browser and searching the link
to events at OpenStack. I am in the event web page and can see PTG
2019, will take place in Shanghai, China on November 4-6, 2019. I
can see the Venue, duration and format. Next, I am scrolling down
and can see the technical detail defining a PTG, financial
support and accommodation and previous PTG events.
. . . [hmm] Interesting! I just found an exciting text here it reads
"(PTG) provides meeting facilities allowing the various technical teams
contributing to OSF projects (be it code, documentation,
operator or user feedback) to meet in-person, exchange and
get work done in a productive setting."

- :P29:
  - TASK: Create a feature branch on your project and push it upstream.

OK. I am cloning the Glance repo and git checkout -b [BRACHNAME] naming the feature branch awesomeness. I am not implementing concrete features but small commits and make sure

I am ahead of the master. I am making my first commit. ... typing staging it and push upstream, I am making second commit staging it ... and push, and another commit ... staging, pushing ...

Now I am merging awesomeness to master making the head point to master. .... Resolving conflicts ... done.

Table 6 (P31 ... P36)

- TASK: How can you explain the Depends-On Tag used on multiple project repositories?
- observation :P30:

  This tag works on changes that happens in multiple project repositories,
  which developers indicate the dependent patches with the 'Depends-On' tag.
  There the tag will appear as a link in the commit message similarly to
  the change-ID, this tag helps to track all the dependencies of your changes.
  I am now opening a sample commit message on Gerrit to show you the
  depends-on tag, I am opening a new tab on my browser ... then going to OpenStack Gerrit ...
  search for key-word depends-on .... I found couple of them, I am selecting anyone at random...
  I am opening the change set ... this concerns the Nova compute and has
  many depends-on I can't count all for now
  What else am I missing out Um . Also, patches cannot be merged unless all its dependencies are landed.
- TASK: Can you create a patch with the depends-on tag? While doing please think aloud.
- :P31:

  Let me create a new feature that depends on four cross-project teams. I am naming it four-factor... OK I am using the template for new features that OpenStack proposed, I am writing

the description ... now, I am using the tag and mentioning all projects that this feature will depends on. oh-oh that doesn't seem right at all. I make a mistake here, let me correct it... Listing project names ...

OK looks good now, let me push the feature.

Table 7 (P37 ... P42)

- TASK Issues/Task tracking : Select on of the bug IDs relating to cross-projects, using launchpad and track all the projects that are reported as affected by the bugs.
- observation :P39:

  I am choosing the first bug-id and searching the bug on launch-pad...
  Now I can read what the bug is all about and the status... Next, I am searching how many projects have reported this same bug
  .... It seems three projects identifies with this bug but not link to this bug directly.
  I am not checking the documentation to see if some of the tags I see here have special meaning .... ok based on the page on Bug Tags,
  Next, I am reading ... OH I see that each some tags are general to all projects and other tags are specific to each project, [Hmm]
  this is a problem now because I can tell if a different project used a different tag to mark this bug. I don't know how to handle this anymore.

- TASK Issues/Task tracking
- observation :P40:

  P40 First, I am using my favorite editor vim to create a python file.
  I will call it exercise1.py Next, I am writing a function that reads
  the prints of all OpenStack summits and their locations in the past

.. goes silent for a while... Now, I am injecting a bug to my code with
the tag Bug101. This bug is assigning the wrong locations to each
summit. I am done with the code. I am adding it to my stage area,
... Now, pushing it to the sandbox repo ... done!Now, I am signing in to
launchpad ... I am reporting the bug now and assigning it to myself ...
the bug is now assigned to me.

Table 8 (P43 ... P48)

- TASK: How can you test your full stack deployment
  - P43:

    I will run the full stack command on DevStack .. let me run the command now on my terminal ... tox -e dsvm-fullstack ... as you can see a couple of messages are displays now on the screen and I am not an expert to interpret things to you, but I am looking of the test will fail or not.
    I read from the docs that full-stack tests often require the same resources and dependencies as the functional tests, reason why I used the configuration script tools/configure$_{\text{forfunctesting.sh}}$ ... So, I am adding a log file /opt/stack/logs/dsvm-fullstack-logs/test$_{\text{example.log}}$), so that will be a good place to look if your test is failing.
    Taking longer than I expected ... finally done, let me look into the log file for output messages .... OK looks good

observation :P6: OB1 asked P46 to think aloud while performing the task. P46 "I am writing the body to have all the essential elements [...], I have 120 characters I am now worrying about reducing the number of characters to 72 [... thinking ...] let me play with words a bit to see what I get here [...] The best I can do is 98 characters.

I am now posting this body to the IRC channel ... waiting for mentors' feedback [groans]I made a mistake .... I will have to amend this commits, before someone sees it. .... I will just do a short message type with git commit –amend -m "an updated commit message" ...

Table 9 (P49 ... P54)

- TASK: Getting to know your projects

OB1

What project are you choosing and why? observation :P50: I am searching for Nova, the compute service for OpenStack because earlier today I read about the documentation and I am interested in this kind of projects.

Now, I am cloning the nova repo [git clone ...] git is cloning the repo to my locally, it's [seems a big project] OK done cloning. let me CD into the git repo and explore the content. OB1 it seems your're pretty familiar with git [hahaha, yes that is true. Of course I used git and GitHub at school a lot in almost all my school projects and assignments.] OB1 what are you doing right now? I am looking for the different modules, there are tons of them [wow!] such a huge project and I don't know where exactly to start exploring.... OK I figure out, M9 told us last time to start with what we are most confident about before looking at more complicated things ... I am looking the coding style, code commenting and messages that dev wrote to explain their work ... I am also reading the doctring to know of functions return something ...

That is pretty much it.

- TASK: Can you think aloud as you do the test coverage exercise?
- :P54:

  I am looking for an open review .... I found one, now reading the type of patch
  ... Ah. seems a complicated task, let me find another one that could be more simple to understand ...

ok let me check this ... OK I am downloading this patch... next, I am removing all the code changes .....
now. I am running the test command ...
starting runing and observing the output ...
BAM! it fails. looking for failure message ... I am writing a comment now on the open review .... done

Table 10 (P55 ... P60)

- TASK: review task Review a patch set and provide feedback

  In the open review system, I am looking for patches that need review... I am picking the one with four related projects ... I am reading the commit message ... what was the problem,... OK .. and how did the author of the patch anticipate to solve the problem, ..., and how did the author solve the problem,... Now, I have an insight of the patch set so I am giving my feedback because the message describing the patch is well written and easy to follow.... I think I should run the test cases to see if it passes or fail... download open my terminal tab run the test command .... observing ... good! it pass all the test cases.

observation :P55:

- TASK: Logging
  - Can you place logging in sections of your code and explain what you're doing?

  First, I am using the LOG.debug() in this project – Swift that I cloned I am entering the repo and looking for any source code that I can insert logging .... OK I will make some failure cases since I choose the debug mode ...I am breaking the code .... now i am adding logging.. when it fails the log will tell me what when wrong.

  Now I am running the code ... failed .. now, I am opening the log file to see the messages .... reading error messages ... voila .. as expected .

Table 11 (P61 . . . P66)

- TASK: look for the most common cause of job failure

observation :P62: ZUUL CI/CD jobs

> Checking on ZUUL, going to the web portal . . . navigating to jobs and looking to failures, then observing the failures . . . trying to classify the most occurrences and count them. . . I have found four common causes of failures in ZUUL on the project that I selected and posting the result on IRC channel.

- :P66:
- TASK: DevStack
- Use five openstack commands on DevStack environment and comment how you are doing it.

  I am spining up DevStack in my local environment . . . I am running the command: systemctl status devstack@* this command ensures that devstack is up and running smoothly, now I am using the journalctl command to view output from network service Now, I am copying the openstack commands and paste them on my devstavk environment terminal and observing what each command is doing . . . I am doing the last one and referencing the docs all command ran as expected.

Table 12 (P67 . . . P72)

- TASK Testing exercise P67

OB1 Can you do the exercise on Squashing Changes and explain what you're doing?

> I have made several small commits as the mentors asked us, Now, I want to squash them so that they do not show up in the public repository since each commit is a change in Gerrit, and must be approved separately.
>
> First, I am checking out to the master branch git checkout master
>
> Then, I am pulling the latest changes git pull origin master

> Then I am switching to a new branch I name squash git checkout squash Next, I am rebasing git rebase -i master In am using VIM editor to squash commits that should not appear in the public history. Last, I am preparing the public commit message in my VIM editor. I am starting with the commit message from the commit that I picked, and it should have a Change-Id line in the message.
>
> Done!

- TASK: Use storyboard to track the first bugID that mentors posted on the screen and think aloud as you do it. Find all related projects affected by the bug.
  - :P72:

OB1

> going the storyboard URL . . . search for the bug ID .. navigating the storyboard interface and environment . . . now search for the bug . . . [OK] looks much more simple than launchpad, I can see all the affected project linked automatically to this bug we did this exercise on launchpad and could not see any affected project.