Thank you very much for taking me into consideration for this position.

1. I first check the scope of the newly established commercial claim.

And design the business logic related to it and check the logic and related relationship that has been realized so far.

Technically, the entire identified new part logic is subdivided according to the part to be realized, tech-stack wise, and the temporal and logical order of the algorithm flow.

Next, tickets are created according to each item of smaller tasks. At this time, specific explanations and criteria are accurately described so that developers who will be in charge of them can start working with a correct understanding of the part they need to work on. . After assigning a ticket to an assignee, let him or her understand the correlation with tickets currently applied to other developers through a one-on-one meeting or a simple chat as needed.

1. In general, after a ticket is assigned in the agile/scrum process, the developer creates his/her new branch and performs his/her work there. At the daily standup meeting, find out under what circumstances each developer is currently working and what factors are blocking their development process, and ensure that they continue to work in their ticket in the right direction, and every sprint based on the story point corresponding to each ticket. Have a task-grooming meeting and make sure that there is no pause in performing the ticket task. After once a developer moves his ticket to in review column, review the PR and leave relevant comments for him to eradicate, and use tools such as Jenkins will be used to trigger automatic testing and monitor the result.
2. I am familiar with both mono-repo project configuration and multi-repo project configuration.

Both methods have their advantages and disadvantages.

And speaking of code standards, since we don't have any of them, we can distribute documented standards to developers, or simply create a simple code example snippet that reflects standards in all cases first and have all developers follow it.

In addition, at every reviewing occasion in progress, it should be inspected whether the defined code standard is violated or not followed.

1. If he has 1 year of experience in my team, he will know the logic of the product to a certain extent, so first, I will make sure he correctly understands the relationship between the logic of the task he is performing and the current overall logic.

Then, if he struggles with the practice, I have a one-on-one meeting with him, have him share his screen, and have him try it himself first.

Next, from the part he knows exactly, or the part he can't go any further, I will indicate to him what to do one by one and let him try it with his own hands.

If I come up with some flaws from him when I must review and mentor his results, rather than criticizing those, I'll ask simple questions that lead him to identify the flaws in his code himself.

This way, when he completes his assignment perfectly, I will recheck it and approve it.

Thank you again.

Looking forward to hearing back from you soon.

Best,

Nick Y.