

3 Steps to Completing a PBI (for a developer's eyes only)

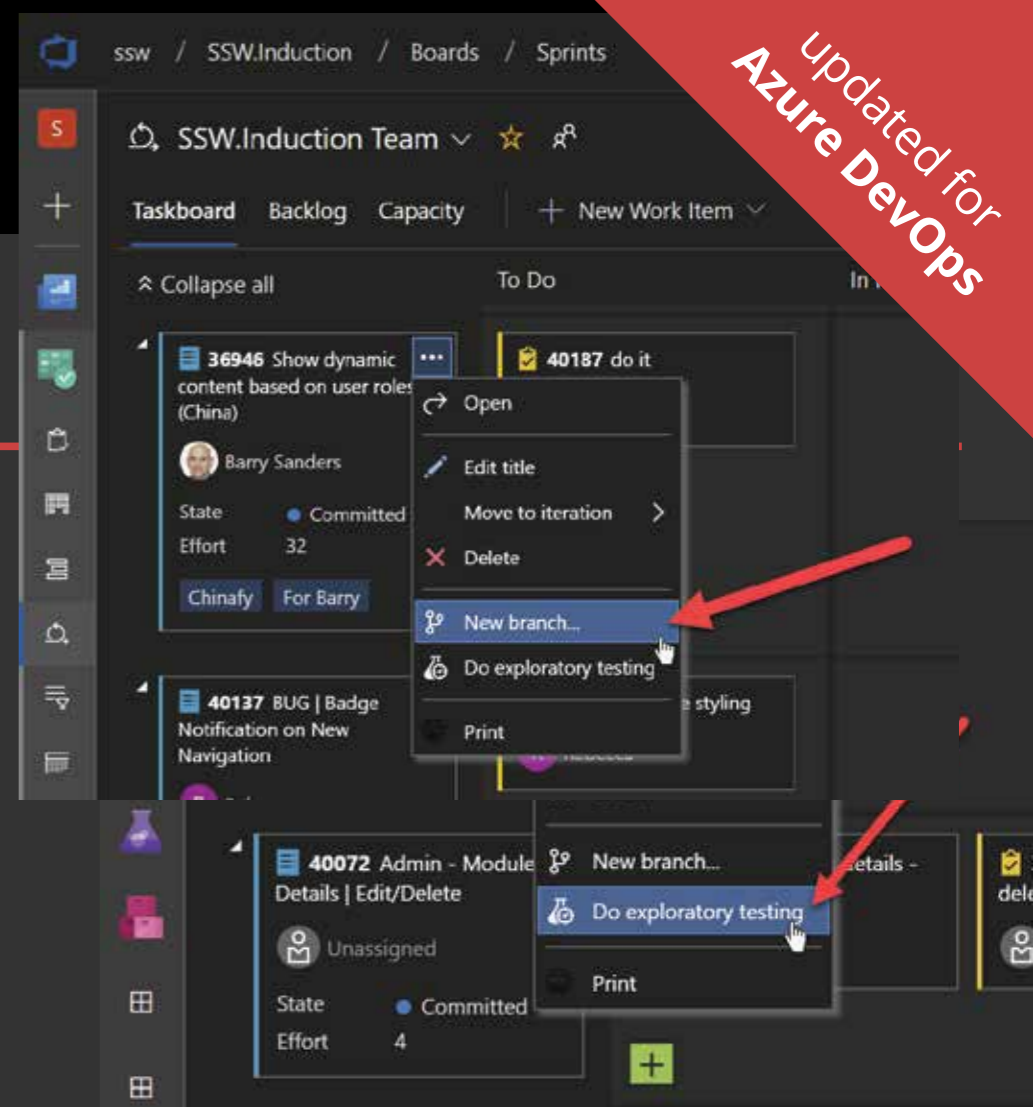
Updated for Azure DevOps

1 READY

1. Take the next PBI that was approved by the Product Owner
2. Is the PBI "Ready"?
Check your PBI against your Definition of Ready. "Ready" PBIs must have Acceptance Criteria. A good Definition of Ready also encourages a test first mentality in requirements e.g. Use Spec Flow (Given, When, Then) and/or create Test Cases and Test Steps first.
3. Attach and copy the Product Owner's email into the Acceptance Criteria (as per Do you attach and copy emails to the PBI?).
4. Email notification to the Product Owner - use the @ mention rule
5. Break down your PBI into Tasks
6. Don't forget to make a Task for testing!

2 CODE

1. From the PBI, create a new branch (so that your work is automatically tagged to the PBI)
2. On the Kanban Board, move your Task into "In Progress"
3. Checkout your new branch and start coding!
4. Code, code, code... (Red, Green, Refactor)
5. Push your changes, open a Pull Request
6. If you want feedback early, record a "done" video
E.g. Snagit



3 DONE

Is the PBI "Done"? Check your Definition of Done, and then the Tester:

1. Open a Pull Request
2. Get another engineer in your team to do an "over the shoulder" check of the code
3. Use Microsoft's "Test and Feedback" Chrome extension to test the app
4. Make changes based on feedback (and then get more feedback)
5. Complete the Pull Request! Merge changes to master, this automatically deploys (to either Test, Staging or Prod based on process maturity)
6. Email notification to the Product Owner – send a Done Email. Big items should have a Done Video.

Congrats. Your PBI is now ready to be demonstrated during your Sprint Review!



Let's take a look at what happens to a task in a sprint!

4. Sprint

