Software Development Life Cycle (SDLC)

1. Requirement Gathering:

1.1 BA gathers requirements from the client

- What type of product is the client looking for?
- Within how many months they want to get it delivered?
- What are the critical concerns for the client in terms of testing (Security testing, database testing, functional/application testing, API testing)?
- BA and client have a clear understanding of the requirements.
- BA and client have a clear agreement on what will be delivered as a part of MVP.
- Clients need to confirm if they want UX/UI developer or they need to provide the proper template in which they want to see any functionality.

1.2 MVP

- Testing will be functionality and integration specific. UX/UI related things, Minute details (in terms of how logo is, wordings of para, fonts, Headings etc.) can be avoided as a part of MVP. This type testing can help to deliver a sound minimum viable product while still respecting cost limitations.
- If client is particular about UX/UI stuff, need to have UX/UI included from the beginning of the project
- If a client wants to make an app browser compatible as a part of MVP, then it will require more time.
- Data from the client (if that is an existing data), need to be in correct form. No duplicates, no junk data. Not doing so, affects the application in many ways and continuously making changes to the data makes development and testing difficult.

2.Requirement Analysis:

- BA set-up a meeting with the developers, QA, PM and other project related stakeholders
- BA explain the requirements to the team and makes sure that its clear to all
- There needs to be a mutual agreement and a doc needs to be created by the BA/PM stating the MVP expectations.
- BA needs to create a requirement gap document, which is based on the doubts and questions raised by the team and same need to be answered by the client before development starts.
- Team (Most of the time its dev team) will decide and estimate how much time each requirement will take to get completed.

3.Designing and Planning

- Architecture will be created for the application which will include

1. The database or data warehouse which will be used for the applications data storage
2. API setup
3. The complete flow of the application
4. Different environments for development and testing (Dev, TEST, Staging, pre-prod, prod etc.)
5. A complete architecture detailing the project (storage, environments etc.) which will be provided to client and team
6. ü Separate databases for DEV/SIT and UAT environments

- QA/PM will create/confirm workflow for the development and testing process which includes Jira tickets workflow and Sprint board workflow.
- QA and PM will work together to create a test plan and test strategy.

3.1 Product Backlog

- PM creates Product backlog based on the client requirement
- PM prioritize Product backlog depending on how critical/severe the ticket (epics, user stories, sub-task, bugs) are.

- Tickets need to be moved from Product backlog to the current sprint during the sprint Planning meeting based on the team discussion

3.2 Sprint Planning

- Sprint Planning meeting need to be done only in the starting of each sprint
- Daily standups are done where team share:

1. What they did yesterday
2. What they are doing today
3. If anything is blocking them.

- In the starting of sprint planning, we need to highlight what we will be delivering at the end of the sprint to the client
- In one sprint we need to focus on counted number of tickets not altogether.
- Everyone needs to work on the same ticket at the same time so that we can deliver some completely working, stable and bug free feature to the client
- If bugs are reported, they need to be fixed in the same sprint. Only if the bug is of low priority then it can be moved to another sprint.

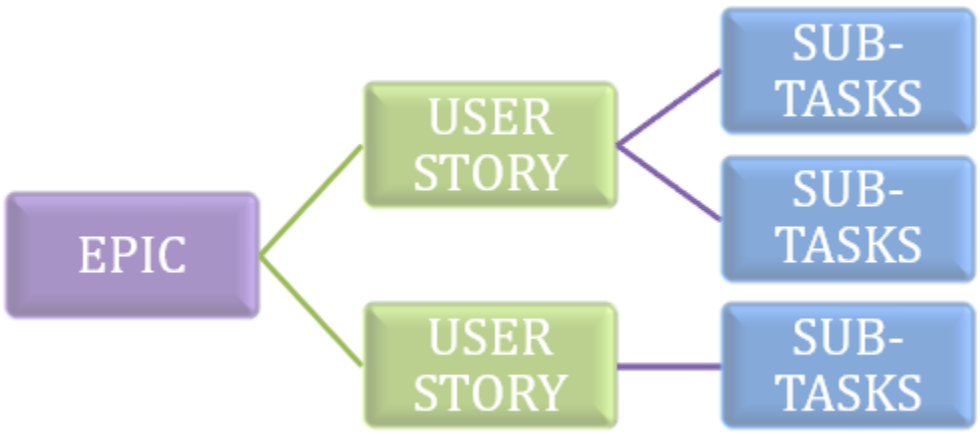
4.Development and testing

- PM will update the sprint backlog on which developer and QA need to work

1. Detailed user stories will be created by the PM and Mock images need to be attached to the tickets for easy understanding
2. In refinement, requirements will be made clear to the team. The approach and the estimation of time need to be discussed

- QA creates ACs for the Ticket and assign to PM for the review
- Once Confirmed by the PM, QA creates scenarios/TCs
- QA then assign ticket to developers
- Once development done, ticket is assigned to tester
- Testing is done, passed to PM for final review and then moved to UAT
- Once the complete feature(which is the part of current sprint) is developed, successfully tested and delivered to the client then only automation will be done for that feature (Automation will be covering only the happy path to ensure that critical functionality is working fine)
- Clients need to do testing every time a new deployment is done

4.1 Requirements Breakdown



EPICS	USER STORY	SUB-TASKs
Major Feature/functionality	Epics are divided into user stories	If new requirements are added by the client then that will be added as a sub-task to the user story
Can take more than 1 sprint	Need to be completed withing the sprint	
One line description	Detailed description of what is required by the client	Detailed description of what is new feature, how old features are getting affected, what all other tickets are getting affected by this new requirement
Need to have all the user stories attached to it	Need to have all the ACs, TCs, Sub-tasks, bugs, mock-ups/templates attached to it	Need to have bugs related to this subtask attached to it

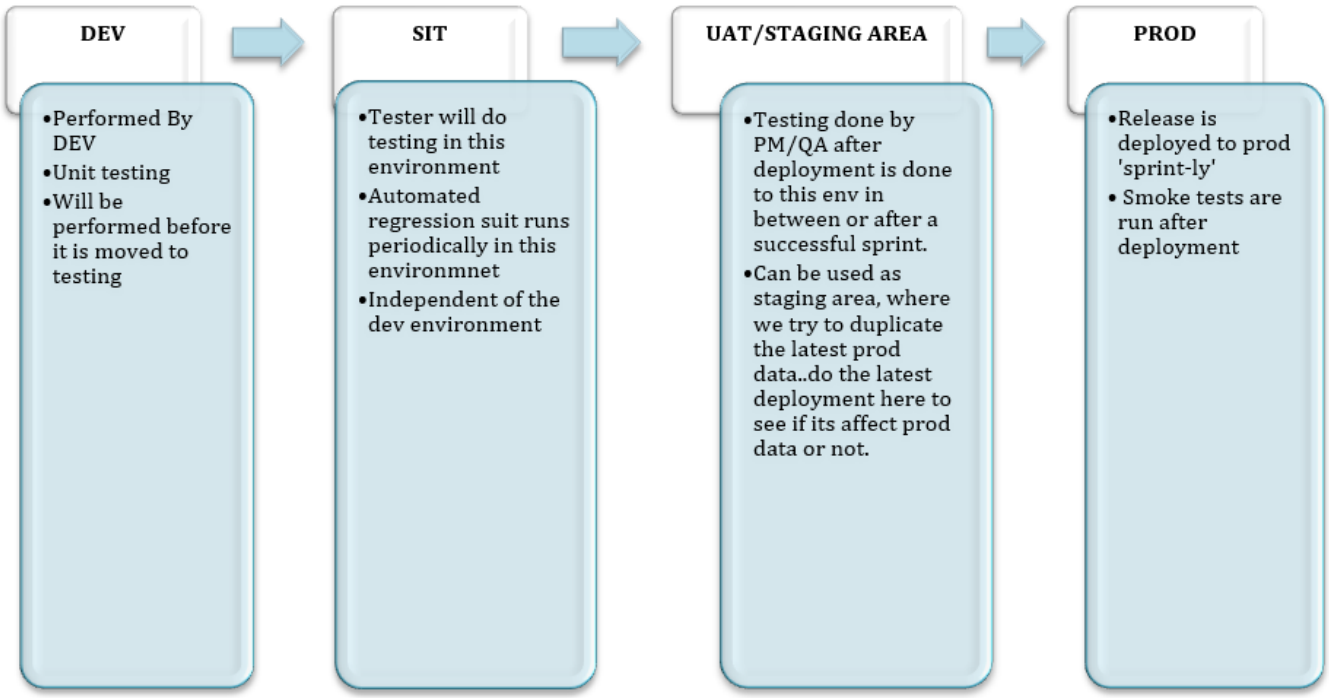
4.2 Sprint Board

BLOCKED /HOLD	TO DO	IN PROGRESS	READY FOR TESTING	IN TESTING	IN UAT/PM REVIEW	RELEASE TO PROD
Blocked by other ticket	Will be completed in current sprint	Dev team is working on it	Ready for QA	QA is testing the ticket and is in progress	Tickets has been passed by the QA and PM need to review if everything is as expected	Ticket is thoroughly tested and verified and can be now implemented in the PROD/LIVE environment

4.3 Test Case management

This can be done with the help of TM4j (now called Zephyr) addon.

4.4 Test Environments



5.Deployment or Release

o Deployment to the testing env, need to be done only once in a day (unless there is any blocker):

1. Not doing so, make site down multiple time, affecting testing time
2. Can introduce to new bugs which will further affect the testing

o Release to the UAT/PROD needs to be done at the end of every sprint.

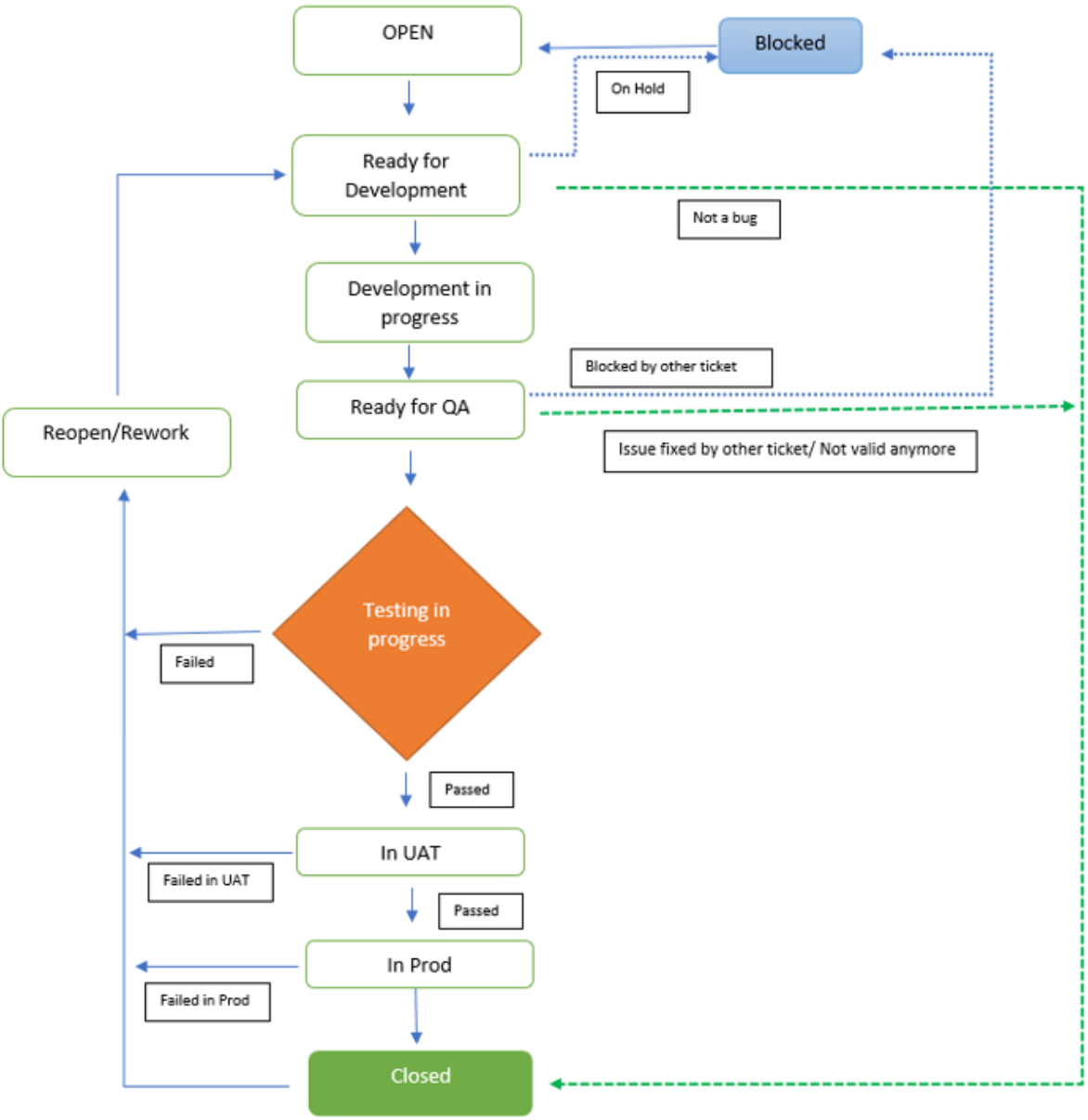
o Once deployment/release has been done, release notes are prepared by the PM highlighting the tasks which are completed/resolved or if there is any blocker

o Clients can be asked to test the latest deployment changes after 1 or 2 days of actual deployment. In these 2 days smoke testing by Born can be done, giving enough time for fixing issues which are introduced due to new deployment.

o Issue/new requirements logged by client need to be included in next sprint, unless they are blocker or of high priority

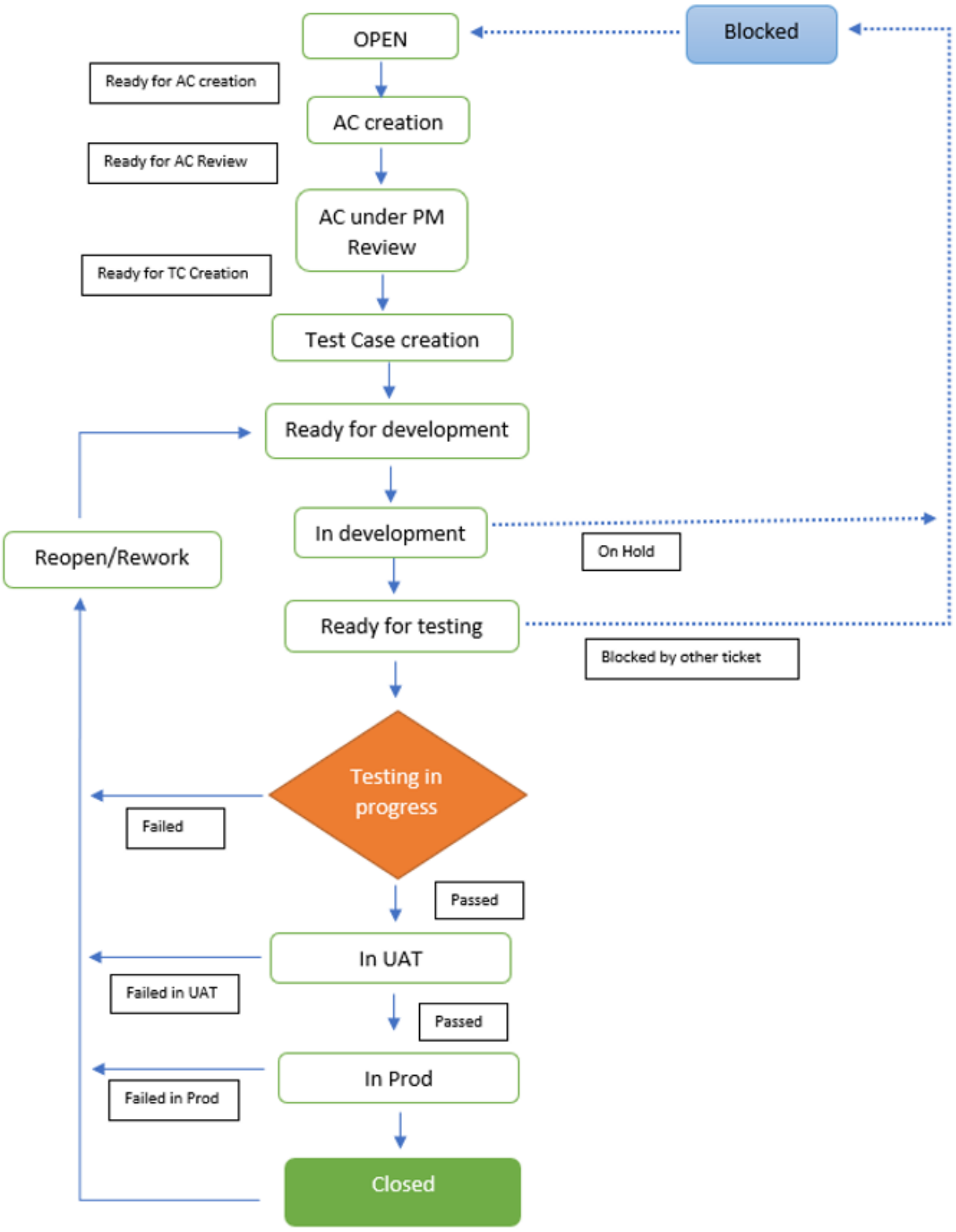
6.Workflow for different tickets

6.1 Bug workflow



6.2 Epic workflow

6.3 User story workflow



6.4 Sub task workflow

It has same Workflow as User story.