Project standard going forward:

1. Select Classic Project when setting it up on JIRA.

A picture containing clock

Description automatically generated

1. Enter board name and Choose Scrum board

A screenshot of a cell phone

Description automatically generated

1. Board layout:

* Product Backlog - All features will go into the product backlog. These will be added on the backlog page on JIRA. From there, they are pulled into the ready column(when they are ready) so we know they are ready for development.
* Sprint Backlog – items that are ready are pulled into the sprint backlog once we’re ready to create a sprint. These are the items that will be completed in the sprint.
* In progress – this is work that is currently being worked on.
* Done – items will be moved to the done column when the code is complete.
* Done done – items will move to done done once the code is complete and the feature has been fully tested.
* Below is a visual example of what our boards will look like:

A screenshot of a cell phone

Description automatically generated

1. All issues need the following:

* Name
* Assignee
* Reporter
* Story points (see number 4 on how allocate story points)
* Priority (Although a backlog is already prioritized)
* Time estimated according to story points (1 point = 1 day. 0 points means it’s really quick)
* Issues need to be allocated to the correct issue type; task, bug, etc.
* All issues need to be broken down into sub-issues

1. How to assign story points to tasks.

* The bigger the story points for a task, the bigger the task and the longer it will take to complete.
* We will be using the Fibonacci sequence for story points. 1 point = 1 day.
* The numbers we will be using are: 0, ½ , 1, 2, 3, 5, 8, 13, 20, 40, 100.
* 0 story points means it’s a really quick task. However, lots of zeros can end up becoming a ½ or a 1 so 0s need to be accurate.

1. User stories

* All features need to have user stories
* This is the format of a user story:

|  |  |  |
| --- | --- | --- |
| **Title** | **Priority** | **Story Points** |
| Post a photo | High | 13 |

|  |  |
| --- | --- |
| **As a** | user |
| **I want to** | Post a picture |
| **So that** | People can view it |

Business rules:

* eg: Only admins can post pics

Acceptance criteria:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Given** | **When** | **Then** | **Notes** |
| **1** | I am on the create post screen | I click create post | I must upload a photo | Wireframe |

* User stories will be rejected if the dev team do not understand them and they will need to be rewritten.
* User stories must have all fields completed
* User stories will be written by a collaborative effort of the entire scrum team (Scrum Master, Product Owner, Developers)

1. Sprint Planning

* Sprint planning is a Scrum event whereby 2 sprints of work will be planned
* Sprint Planning can take anywhere between 4 and 8 hours.
* Only items that the team believes they can complete in a sprint will be added to a sprint. This ensures that sprint goals are always met.

1. Potentially shippable product:

* The work of the entire team must be integrated before the end of every Sprint—the integration must be done during the Sprint.
* Potentially shippable means that the feature works in it’s entirety but it doesn’t mean the entire product is finished.
* An example of a potentially shippable product would be a login. The login works, authentication works, etc. But without the rest of the project (after you’ve logged in) there is no value to the end user. Thus it is potentially shippable.

1. Scrum events

* Sprint Planning
* This is the event that kick starts each Sprint and is where the Product Owner and Development team discuss which Product Backlog Items (PBI’s) will be included in Sprint.
* The outcome of Sprint Planning is to get a Sprint Goal and Sprint Backlog that everyone agrees is realistic and achievable.
* Daily Scrum
* (Stand Up)
* Needs to be timeboxed to 15 minutes
* The Daily Scrum is an opportunity for the Development Team to check in, assess progress towards achieving the Sprint Goal and to review and plan their activities for the next 24 hours.
* Sprint Review
* This takes place at the end of each sprint.
* Work that is done done can be shown to the product owner and stakeholders. This work can even be signed off by all concerned parties such as dev team, stakeholders, product owner, project managers, etc.
* Sprint Retrospective
* This is when the Scrum team reviews what could be improved for future Sprints and how they should do it.
* No matter how good a Scrum team is or can get, there are always ways to improve.
* The Sprint
* We will be doing 1 week sprints instead of 2 week sprints.
* The purpose of having 1 week sprints is to shorten the feedback loop and also to gauge velocity earlier on in the project.
* The Sprint is an event in itself that contains all the work and all the other events that happen during the time boxed period of development.

1. Scrum artefacts

* Product Backlog – prioritized list of all work that needs to be done
* Sprint Backlog – Product Backlog Items that will completed in the sprint
* Product Increment – The done piece of work at the end of a sprint.
* Product backlog refinement – happens during the sprint. The Scrum team decides when this will be done. This insures that the product backlog is always prioritized and up to date.

1. Agile manifesto

* **Individuals and interactions** over **processes and tools**
* **Working softw**are over **comprehensive documentation**
* **Customer collaboration** over **contract negotiation**
* **Responding to change** over **following a plan**

1. Definition of ready:

* A user story must be immediately actionable. It means that a specific product backlog item (PBI) is ready to start development
* This means there are no impediments preventing this PBI from being ready to dev.
* A ready PBI is one that has a user story, adds value to the end user and is good to go.
* Only items that are ‘ready’ will be pulled into the sprint backlog

1. Definition of ‘done’ and ‘done done’.

* ‘Done’ means that the code for that feature is completed.
* ‘Done done’ means that the code is completed, and the feature has been fully tested.
* Our goal here is to get everything in a sprint to ‘done done’.