|  |  |  |  |
| --- | --- | --- | --- |
| **Need** | **Priority** | **Features** | **Plan Release** |
| Create Different Role |  | (1) Different Roles like Product Owner, Scrum Master, Developers, Tester, and Executives should be created and stored in the system. |  |
| Customer can submit/update requirement |  | (2) Customer need to login to the system to access the application.  (3) Customer can insert requirement using an online form and the application store the requirement.  (4) Customer can also update his/her requirement online using update form. |  |
| Create/Update User Stories |  | (5) Approved user login to the application and can create user stories and store it.  (6) Approved user can also alter the stories and store it. |  |
| Create/Update Product Backlog |  | (7) Create Product Backlog using all user stories and store in DB.  (8) Can update the backlog anytime and store in DB. |  |
| Identify user stories for release backlog |  | (9) Check the required user stories for the system and give priority to each story to create a release backlog. |  |
| Create/Update Sprint |  | (10) Create Sprint by selecting user stories from release backlog and store it.  (11) Can also update each Sprint before finalize them. |  |
| Scram Master can assign job, setup meeting, release planning |  | (12) For each Sprint Scrum Master  will assign job to developers |  |
| Approved developer can submit and update estimates for each item of a given sprint |  | (13) Approved developer can login to the system and see jobs assigned to them.  (14) Developer insert the amount of work involve for each item in hours and days and submit the form. |  |
| Monitor each Sprint |  | (15) Provide a visibility tool to monitor the progress.  (16) Provides day by day measure the amount of work remains in a given sprint or release.  (17) Calculate the Burndown velocity |  |

Role Creation [Product Owner, Scrum Master, Developers, tester, executives]

Customer can submit/update requirement

Create User Stories: User Story: As a (role), I want (feature), so that (benefit)

Customer, executives, developers

Update stories

Create Product Backlog: collection of user stories

Identify user stories – release backlog and prioritize each user story.

Create Sprint by splitting realized backlog

* Date range for each Spring

Scrum Master: - setup meetings, job assignment, monitoring, release planning.

Who is working on what?

Team estimate the amount of work involve for each item

* In amount of hours and days
* 1, 2, 4, 8 [bucket size]
* 2, 3, 5, 10 [bucket size]

Monitor each Sprint

* Using burndown chart [visibility tool]
* Provides day by day measure the amount of work remains in a given sprint or release
* Calculate the Burndown velocity.

Developer:

Assigned developer submit estimates for each item of a given sprint

Update the amount of time remaining for each item

Daily Scrum (follow agile development methods)

* Communication flow between team members
* Meeting