## Jira Process Notes

* **Epic** is equivalent to new functionality that is a large scope of interrelated work that spans multiple Sprints which allows for an iterative / phased rollout of functionality.
* **User Stories** are stories that describe user goals and use cases. User Stories should be written with *the goal* of completing design and development within a 2 week sprint.
* **Sub tasks** are created by the assigned owner of the user story to break down the work into tasks to estimate in order to deliver the user story.
* **Bugs** are created for defects or missing functionality from user stories or tasks.
* **Tasks** should be created for tasks that need to be scheduled and completed that are NOT related to user stories.

## Jira Ticket Workflow

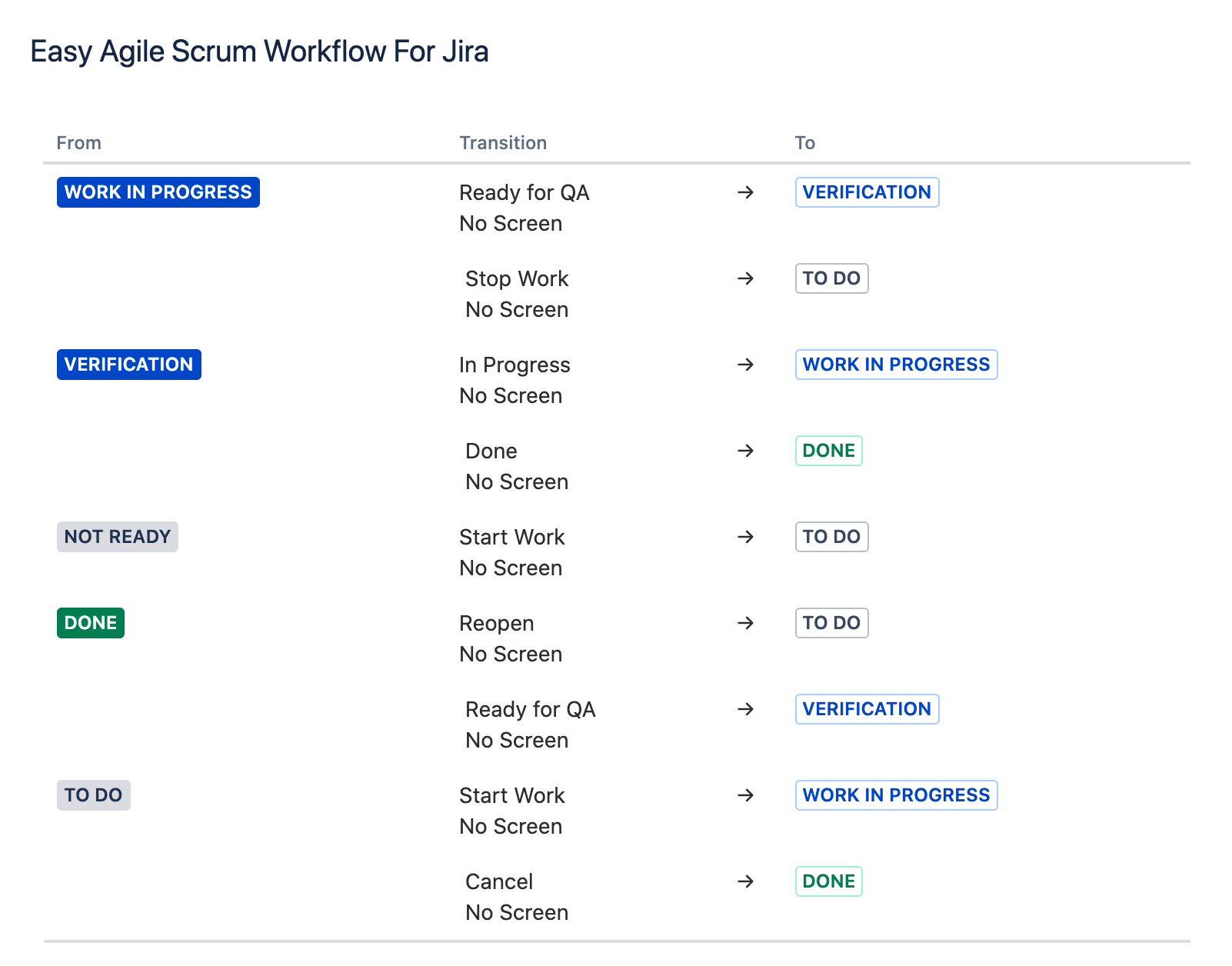
1. Placeholder / Create new User Story
   1. Assign to an Epic; goes into backlog
2. User Story is reviewed and approved for rough estimation (t-shirt sizing in days)
   1. Add / clarify requirements if needed
   2. Define priority and rank order
   3. Ready to move to Sprint for planning
3. Sprint Planning (ideally first day of Sprint)
   1. User Story is completed and ready for final estimate (days and fractions ¼, ½, of days)
   2. Finalize priority and rank order
   3. Ready for implementation and assigned to development resource
   4. Sub tasks are created and estimates entered in days
4. Start the Sprint in Jira
   1. 80% of tickets must be finalized and estimated

## 

## Ticket States:

## View diagrams and text in Jira here <https://togetherseniorhealth.atlassian.net/plugins/servlet/project-config/TOG/workflows>

* Not Ready (for development) > User Story, Acceptance Criteria or designs in progress
* To Do > Ready for Estimation
* Work In Progress > Implementation in progress
* Verification/In Review > Verify what was implemented by code review, QA testing, product signoff
* Done > Ticket or subtask completed
* Blocked: Add “Blocked” to the label field and assign it to the person that can unblock the ticket. Tag @person in the comment field of ticket to explain why ticket is blocked and what needs to be clarified / resolved in order to continue work on ticket. Move back to “To Do” state.



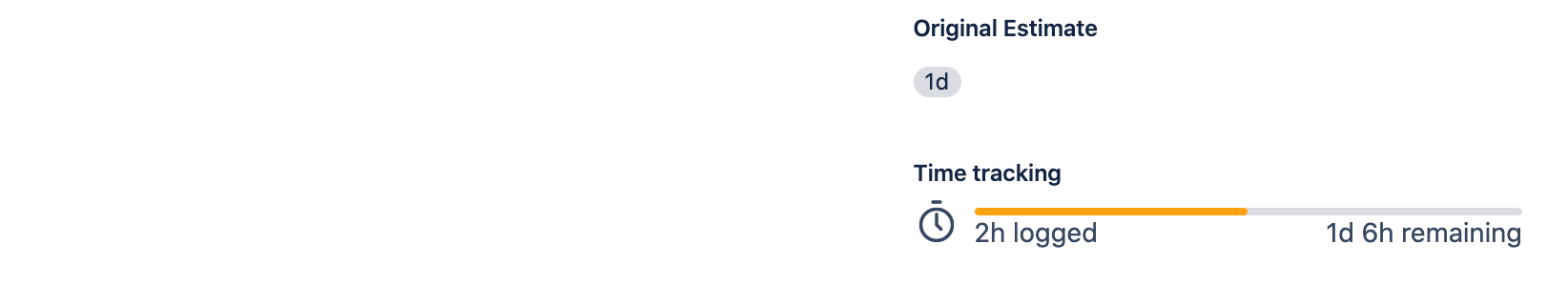
## Jira Components:

See [High Level Component Architecture](https://docs.google.com/document/d/1aYyCwDJMrXQiJ34cSGrDT0xf7PC4k1yAWuRMjZgh45w/edit) document for details

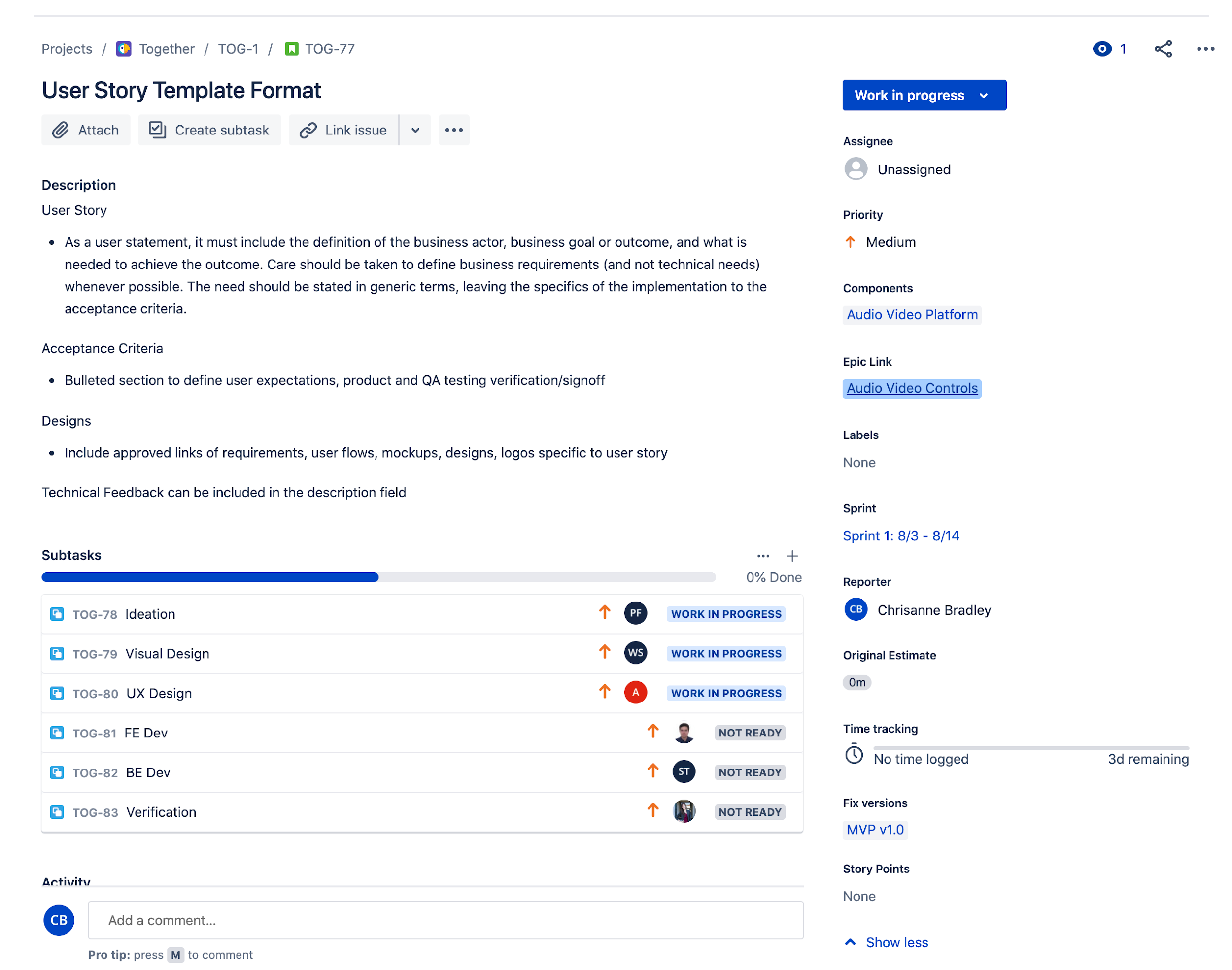
## 

## Jira Ticket Requirements Overview:

User Story:

* High level functional description for title
* Description field must include:
  + At least 1 user story description: "As a..." statement, it must include the definition of the business actor, business goal or outcome, and what is needed to achieve the outcome. Care should be taken to define business requirements (and not technical needs) whenever possible. The need should be stated in generic terms, leaving the specifics of the implementation to the acceptance criteria.
  + Acceptance Criteria section for user expectations, product and QA testing verification/signoff
  + Include approved links of requirements, user flows, mockups, designs, logos specific to user story
  + Design and Technical Feedback can be included in the description field
* Specify Component that user story is related to
* Specify priority and add severity?
  + Define and customize? Highest, High, Medium, Low, Lowest
* Estimates in days must be provided in estimate field (determined during Sprint planning)
* Use comments to communicate changes, questions, clarifications - avoid from editing description field once ticket is in progress

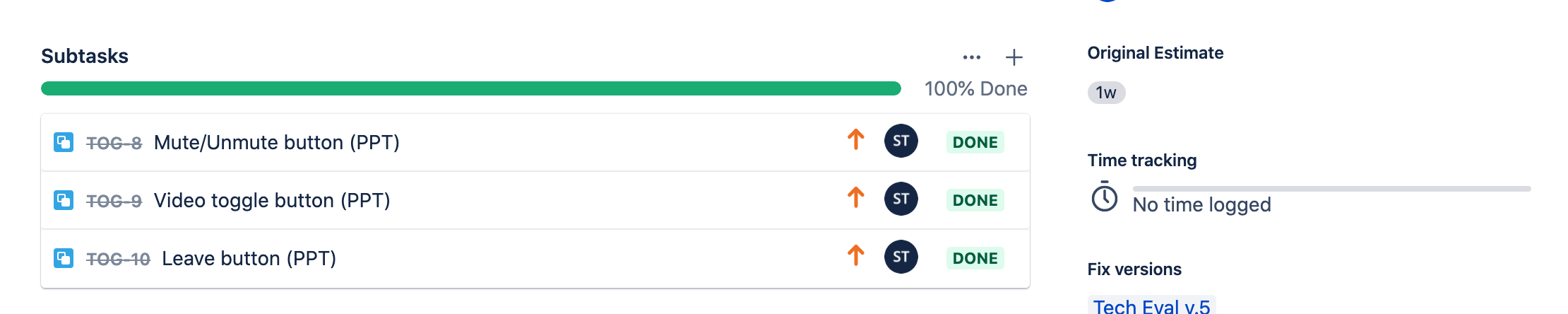
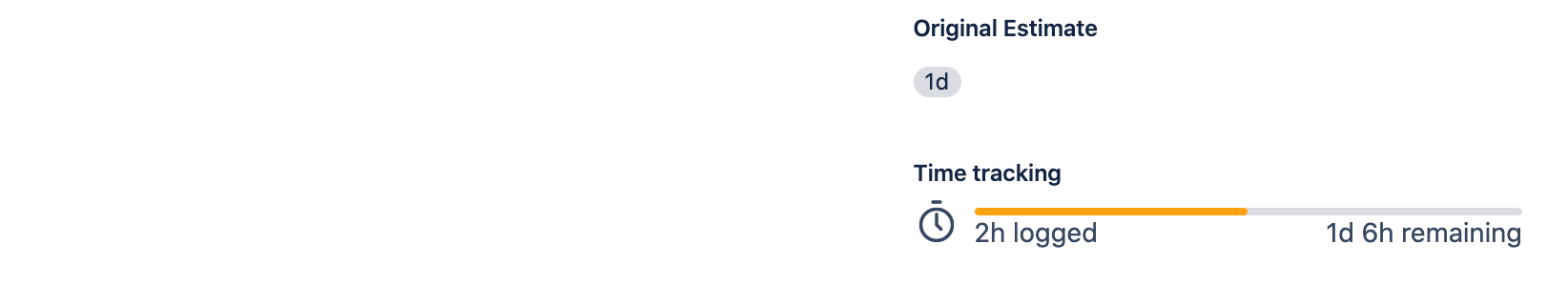
### Example User Story ticket with subtasks - [TOG-77](https://togetherseniorhealth.atlassian.net/browse/TOG-77)



Bug:

* High level description of issue for easy id and triage (include error, epic, component)
* Description Field must include:
  + Prerequisites and steps for test(s) being executed (including system, configuration, devices, version, person, logs)
  + Test steps to meet acceptance criteria
  + Exact steps to easily reproduce (bug must be reproducible)
* Attach errors and logs if available
* Specify Component AND Affected Version that bug is related to
* Specify priority and add severity?
  + Define and customize? Highest, High, Medium, Low, Lowest
* Use comments to communicate changes, questions, clarifications - avoid from editing description field once ticket is in progress
* Assumed default estimate to fix a bug is 1 day; if estimate is > 1 day, enter estimate

Sub Tasks:

* Sub task title should be break down of tasks to deliver user story by a single individual or specific job role
* Example sub tasks: Ideation, Visual design, UX Design Front end Development, Backend Development, Verification
* Estimates in days must be provided in estimate field
* Use comments to communicate changes, questions, clarifications - avoid from editing description field once ticket is in progress
* Sample ticket User Story [TOG-5](https://togetherseniorhealth.atlassian.net/browse/TOG-5)
* 
* Sub task [TOG-8](https://togetherseniorhealth.atlassian.net/browse/TOG-8) related to TOG-5
  + 

## Meetings

* Daily Standups: on video and/or Slack channel > 11:00AM Pacific time
* Sprint Planning Meetings: every 2 weeks; first day of each sprint
* End of sprint demos/playbacks to demonstrate sprint accomplishments
* Design Reviews (as needed, asap)
* Weekly Tuesday evening meeting 5pm Pacific time (Evizi Team)

## Bug Guidelines and workflow

Here are some bug guidelines and Jira workflow recommendations:

* Please create new bugs when verifying/ testing user stories, requirements, etc. (don’t leave comments in user stories if they are clearly issues > create a bug)
* After creating bugs, please link to related user story or task (use “related to” option when linking)
* If you need something (or are blocked) for a ticket “In Progress”, please assign to person you need information from and tag them with @ + name in the comments of the ticket. When the information is provided, the ticket should be assigned back to the requestor.
* Only move tickets to the “Verification” state when they are ready to be verified. If you need something (or are blocked) for a ticket in “Verification” state, please assign to person you need information from and tag them with @ + name in the comments of the ticket. When the information is provided, the ticket should be assigned back to the requestor.
* Note that QA testing and verification should primarily be conducted in the TEST environment. DEV environment should mainly be used by developers or in the rare exception to compare behavior across environments or for a specific request.

## 

## Sample Sprints

Sprint 1:

* Design: Project Kickoff and ramp up on project, research and design the initial set of screens (agreed upon during Sprint planning)
* Development: Project Kickoff and ramp up on project, Agora SDK and existing Angular code
  + Together back-end developer to collaborate with Exygy front-end developer

Sprint 2:

* Design: Design the second set of screens (agreed upon during Sprint planning)
* Development: Implement designs from Sprint 1
  + Together back-end developer to collaborate with Exygy front-end developer

Sprint 3:

* Design: Design the remaining set of screens (agreed upon during Sprint planning)
  + Final design handoff for development
* Development: Implement designs from Sprint 2
  + Together back-end developer to collaborate with Exygy front-end developer
* Verification, Testing and feedback

Sprint 4:

* Development: Implement designs from Sprint 3
  + Together back-end developer to collaborate with Exygy front-end developer
* Verification, Testing, bug fixing and feedback