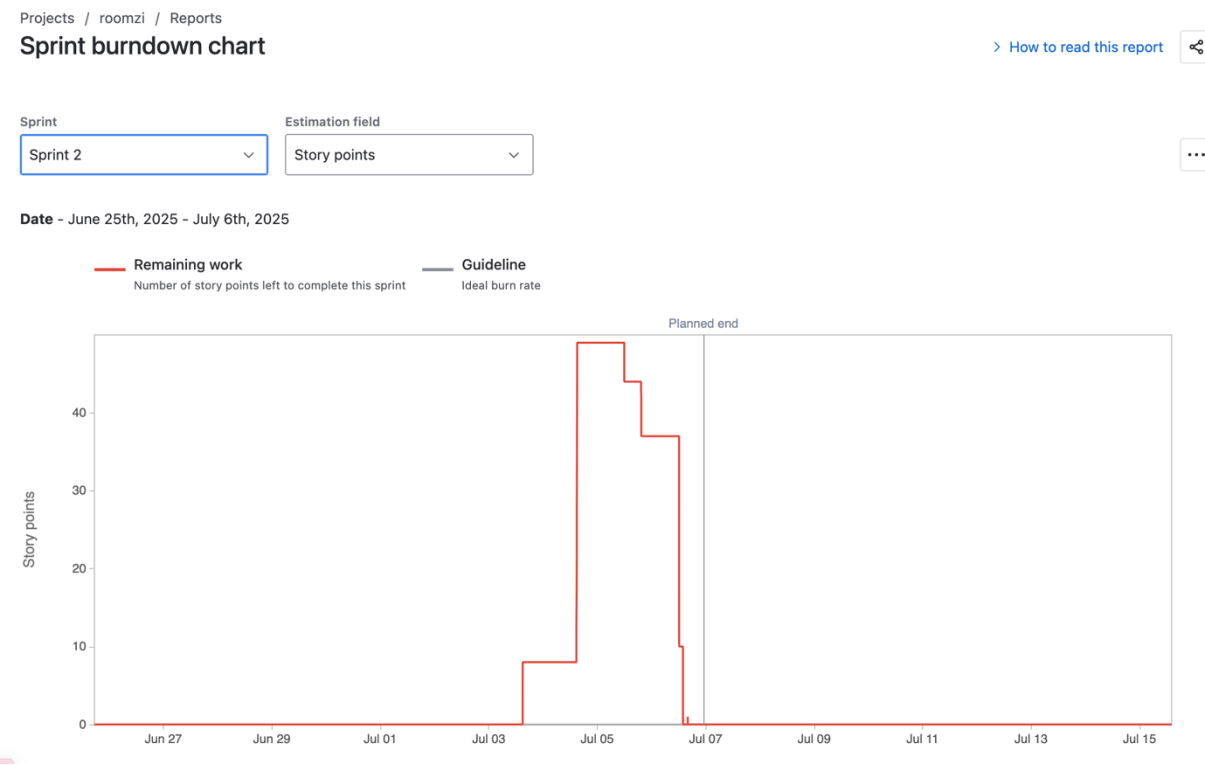
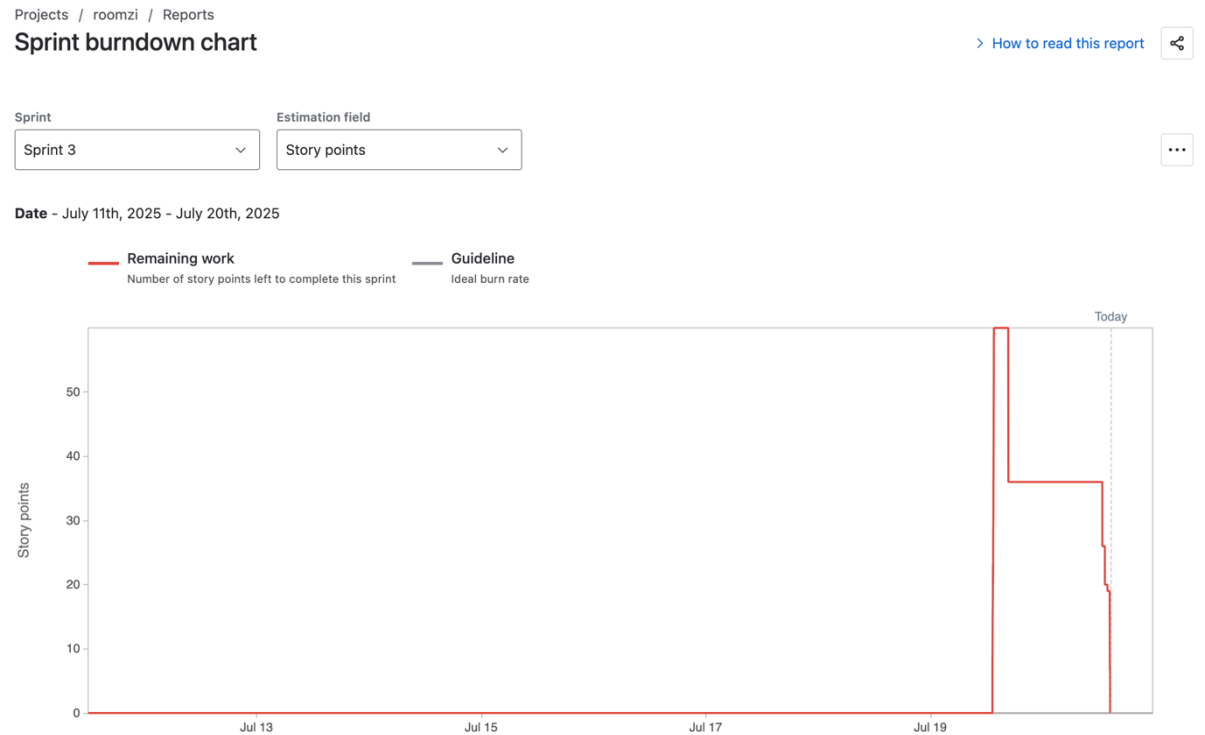


Sprint Burndown Analysis & Reflection

Sprint 2 Burndown Chart



Sprint 3 Burndown Chart



Detailed Analysis

Sprint 2 Overview

- **Dates:** June 25, 2025 – July 6, 2025
- **Story Points:** ~50 points completed
- **Burndown Pattern:**
The chart shows a mid-sprint spike in remaining work, peaking around July 4–5. This indicates new stories were added midway through the sprint. After the spike, there was a fairly steady decline in story points, with all work completed by July 6.
- **Scope Change:**
Sprint 2 involved a significant mid-sprint scope change. New work items were added and some were re-scoped, leading to a visible increase in story points and a deviation from the ideal guideline.

Sprint 3 Overview

- **Dates:** July 11, 2025 – July 20, 2025
- **Story Points:** ~60 points completed
- **Burndown Pattern:**
Unlike Sprint 2, Sprint 3's chart remained flat until July 18, with a sharp drop in remaining story points between July 18–20. This compressed pattern reflects that most development, testing, and task closure occurred during the final two days of the sprint.
- **Scope Change:**
There were several tasks added mid-sprint (document upload, calendar fixes, profile bugs), but they weren't reflected early in the chart. The last-minute updates were a reflection of not updating Jira consistently throughout the sprint.

Sprint Velocity Comparison

Sprint 2 has a mid-sprint spike, then steady decline while Sprint 3 was flat for most of the sprint then had a sharp drop at the end. Sprint 3 saw an increase in velocity, indicating stronger output. However, execution timing and team coordination were less consistent compared to Sprint 2.

Impact of Mid-Sprint Changes

- **Sprint 2:**
 - Mid-sprint re-estimation led to smoother final execution.
 - Despite changes, burndown was controlled.
- **Sprint 3:**
 - Delayed execution compressed work into final sprint days.
 - Task updates were delayed in Jira, skewing the burn pattern.

Changes and Reasons

1. **Increased Complexity in Sprint 3:** More backend/frontend dependencies (e.g., lease creation/renewal, calendar logic) delayed development.

2. **Scope Adjustments:** Bug fixes and enhancements were added as sprint progressed.
3. **Late PR Reviews and Merges:** Much of the work was finalized together rather than incrementally.

Reflection & Lessons Learned

1. **Sprint Planning Discipline:** Scope should be locked after sprint start unless blockers emerge.
2. **Real-Time Task Updates:** Team must consistently update Jira to reflect true burn and help coordination.
3. **Integration Timing:** Spread out backend/frontend integration across the sprint, rather than clustering near the end.
4. **Velocity Growth:** The team has clearly increased its delivery capacity, which could be beneficial for future sprints.
5. **Success:** Despite tight execution, the team completed 100% of Sprint 3's goals, delivering key functionality and increasing project momentum.