# Test Case: Reviewer Assignment Logic (TC\_RA\_005)

|  |  |
| --- | --- |
| TestCaseId | TC\_RA\_005 |
| Test Summary | Ensure fallback to same-department reviewer when no alternative is available, and validate load balancing. |
| Description | This test verifies that the Reviewer Assignment Algorithm assigns reviewers with the lowest current workload while prioritizing department separation. If no reviewers from other departments are available, the system must assign a reviewer from the same department. If no reviewers are available at all, the system must handle this case without crashing. |
| Prerequisite/Pre-condition | - Projects.csv and Users.csv exist - At least one project in 'Overdue' or 'Due Soon' status - Reviewer pool includes:  • A case with reviewers from multiple departments  • A case with only same-department reviewers  • A case with no available reviewers |
| Test Steps | 1. Run `assign\_reviewers` 2. Retrieve assignments 3. Check assigned reviewers 4. Validate load-balancing and fallback behavior |
| Test Data | - Project\_ID: P001, Department: IT - Reviewers:  • U001 (IT, Load 2)  • U002 (HR, Load 1)  • U003 (Finance, Load 0) - Project\_ID: P002, Department: Finance (all reviewers same department) - Project\_ID: P003, Department: QA (no reviewers) |
| Expected Result | - P001 assigned to U003 (lowest load, diff. dept) - P002 assigned to U004 or U005 (same dept fallback) - P003 skipped, no assignment - No errors or crashes |
| Actual Result | Matches expected behavior: U003, U004, and no assignment for P003. |
| Test Result | Pass |
| Automation Status | Automated (test\_reviewer\_assignment.py) |
| Date | 2025-05-13 |
| Executed By | Cynthia McGinnis |
|  |  |

## Revision History

|  |  |  |
| --- | --- | --- |
| Date | Author | Description |
| 2025-05-13 | Cynthia McGinnis | Initial test case for reviewer assignment edge cases and load balancing.  Failed |
| 2025-05-13 | Cynthia McGinnis | Updated logic in assign\_reviewer() and assign\_all\_reviewers() to handle empty reviewer pool safely. Pass |