# CapTrack Use Case Narratives

This document lists the primary use cases in CapTrack in a formal table style that you can copy to Word or export to PDF. The content reflects the current implemented system.

Conventions: Preconditions must be true before start. Success Guarantee states the condition after a successful run. “Actor action” and “System Response” are shown in parallel.

## Use Case 01: Import Students

| Use Case | Import Students |
| --- | --- |
| Actors | Chairperson/Coordinator (Administrator) |
| Pre-condition | Admin is logged in; student Excel file is prepared in required format |
| Post-Condition | Students are created with default credentials and (optionally) enrolled |

| Flow of Events |  |
| --- | --- |
| Actor action | System Response |
| 1. Admin opens Student Import. | 1. System shows file upload form and template guidance. |
| 2. Admin uploads Excel and confirms import. | 2. System validates file structure and row data. |
| 3. Admin waits for result. | 3. System creates student records with default password and reports counts. |

| Alternative Scenarios |  |
| --- | --- |
| Actor Action | System Response |
| 1a. Invalid file or wrong columns. | 1a. Show validation errors; reject file. |
| 2a. Duplicate student IDs/emails. | 2a. Flag duplicates; skip or stop per rule and report. |
| 3a. Admin cancels import. | 3a. Discard upload; return to import page. |

## Use Case 02: Import/Create Faculty

| Use Case | Import/Create Faculty |
| --- | --- |
| Actors | Chairperson |
| Pre-condition | Chairperson is logged in; faculty data available |
| Post-Condition | Faculty accounts exist (teacher/adviser/panelist), must\_change\_password set |

| Flow of Events |  |
| --- | --- |
| Actor action | System Response |
| 1. Chairperson opens Faculty Management. | 1. System shows list and options (Import, Add Manually). |
| 2. Uploads Excel or fills manual form and submits. | 2. Validate data; enforce unique school\_id/email; role in allowed set. |
| 3. Confirms creation. | 3. Create accounts; show success. |

| Alternative Scenarios |  |
| --- | --- |
| 1a. Duplicate school\_id/email. | 1a. Show error and block creation. |
| 2a. Invalid role/format. | 2a. Normalize or reject with message. |
| 3a. Cancel. | 3a. Discard input; return to list. |

## Use Case 03: Create Group

| Use Case | Create Group |
| --- | --- |
| Actors | Student |
| Pre-condition | Student logged in; not already in a group |
| Post-Condition | Group created; student becomes leader |

| Flow of Events |  |
| --- | --- |
| Actor action | System Response |
| 1. Open Group page → Create Group. | 1. Show form. |
| 2. Enter group name/details; submit. | 2. Validate; create group; assign leader. |
| 3. View group dashboard. | 3. Show group dashboard. |

| Alternatives |  |
| --- | --- |
| 1a. Invalid/duplicate name. | 1a. Show validation errors. |
| 2a. Already in a group. | 2a. Block and show message. |

## Use Case 04: Join Group

| Use Case | Join Group |
| --- | --- |
| Actors | Student |
| Pre-condition | Student logged in; not in a group |
| Post-Condition | Student added as member |

| Flow of Events |  |
| --- | --- |
| Actor action | System Response |
| 1. View available groups. | 1. Show groups accepting members. |
| 2. Select group → Join. | 2. Validate capacity/rules; add member. |
| 3. Open group dashboard. | 3. Show membership status. |

| Alternatives |  |
| --- | --- |
| 1a. Group full/closed. | 1a. Show message; abort. |
| 2a. Already in a group. | 2a. Block with message. |

## Use Case 05: Invite Adviser

| Use Case | Invite Adviser |
| --- | --- |
| Actors | Student (Group Leader) |
| Pre-condition | Student is group leader; faculty list available |
| Post-Condition | Invitation recorded as Pending; faculty notified |

| Flow of Events |  |
| --- | --- |
| Actor action | System Response |
| 1. Open Adviser section. | 1. Show searchable faculty list. |
| 2. Select faculty → Send Invite. | 2. Validate eligibility; create invitation; notify. |
| 3. View status. | 3. Show Pending status. |

| Alternatives |  |
| --- | --- |
| 1a. Adviser already assigned. | 1a. Block; show message. |
| 2a. Faculty conflict (e.g., coordinator of offering). | 2a. Deny with reason. |

## Use Case 06: Accept/Reject Adviser Invitation

| Use Case | Accept/Reject Adviser Invitation |
| --- | --- |
| Actors | Faculty |
| Pre-condition | Faculty logged in; pending invitation exists |
| Post-Condition | On accept: adviser assigned; on reject: invitation closed |

| Flow of Events |  |
| --- | --- |
| Actor action | System Response |
| 1. Open Adviser Invitations. | 1. Show pending invites with group details. |
| 2. Choose Accept/Reject. | 2. Validate conflicts; update status; notify group. |

| Alternatives |  |
| --- | --- |
| 1a. Conflict detected. | 1a. Deny accept; explain conflict. |

## Use Case 07: Track Milestones (Kanban)

| Use Case | Track Milestones |
| --- | --- |
| Actors | Student; Adviser (read/review) |
| Pre-condition | Group has milestones/tasks |
| Post-Condition | Task status updated; progress recalculated |

| Flow of Events |  |
| --- | --- |
| Actor action | System Response |
| 1. Open Kanban board. | 1. Load tasks by status. |
| 2. Drag task/change status. | 2. Validate; update task; recompute progress. |
| 3. See confirmation. | 3. Show success and refreshed board. |

| Alternatives |  |
| --- | --- |
| 1a. Unauthorized change. | 1a. Error; revert. |
| 2a. Server failure. | 2a. Show error; restore UI. |

## Use Case 08: Submit Task Document

| Use Case | Submit Task Document |
| --- | --- |
| Actors | Student |
| Pre-condition | Student member; task exists; assignment rules satisfied |
| Post-Condition | TaskSubmission saved; ProjectSubmission mirror created |

| Flow of Events |  |
| --- | --- |
| Actor action | System Response |
| 1. Click Submit on task. | 1. Show submission form. |
| 2. Choose type, upload file, add notes; submit. | 2. Validate file; save TaskSubmission; create ProjectSubmission; update task to Doing if needed. |
| 3. See confirmation. | 3. Show success and counts. |

| Alternatives |  |
| --- | --- |
| 1a. Invalid/oversized file. | 1a. Show validation errors. |
| 2a. Not assigned to student. | 2a. Deny submission. |

## Use Case 09: Submit Project Document (Direct)

| Use Case | Submit Project Document |
| --- | --- |
| Actors | Student |
| Pre-condition | Student logged in |
| Post-Condition | ProjectSubmission saved (type: Proposal/Final/Other) |

| Flow of Events |  |
| --- | --- |
| Actor action | System Response |
| 1. Open Project Submissions → Upload. | 1. Show form (type, file, description). |
| 2. Select type, upload file, submit. | 2. Validate and save submission; show in list. |

| Alternatives |  |
| --- | --- |
| 1a. Invalid file/type. | 1a. Show validation errors. |
| 2a. Cancel. | 2a. Discard; return to list. |

## Use Case 10: Request Defense

| Use Case | Request Defense |
| --- | --- |
| Actors | Student |
| Pre-condition | Group exists; adviser assigned; progress meets threshold |
| Post-Condition | DefenseRequest created (Pending) |

| Flow of Events |  |
| --- | --- |
| Actor action | System Response |
| 1. Open Defense Requests → New. | 1. Show form with defense types. |
| 2. Select type; add message; submit. | 2. Validate thresholds and pending duplicates; create request; notify coordinator. |

| Alternatives |  |
| --- | --- |
| 1a. No adviser / threshold unmet / duplicate pending. | 1a. Block with reason. |

## Use Case 11: Schedule Defense

| Use Case | Schedule Defense |
| --- | --- |
| Actors | Coordinator |
| Pre-condition | Approved request; panel and room availability |
| Post-Condition | DefenseSchedule created; stakeholders notified |

| Flow of Events |  |
| --- | --- |
| Actor action | System Response |
| 1. Open Defense Requests; pick one. | 1. Show scheduling form. |
| 2. Set date/time/room; assign panel; confirm. | 2. Check conflicts; save schedule; notify participants. |

| Alternatives |  |
| --- | --- |
| 1a. Conflicts found. | 1a. Show alternatives; require change. |

## Use Case 12: Provide Feedback

| Use Case | Provide Feedback |
| --- | --- |
| Actors | Adviser/Teacher |
| Pre-condition | Faculty has access to the submission |
| Post-Condition | Feedback saved; status updated; student notified |

| Flow of Events |  |
| --- | --- |
| Actor action | System Response |
| 1. Open a submission. | 1. Show details and feedback form. |
| 2. Enter status (Approved/Rejected) and feedback; submit. | 2. Validate; save feedback; timestamp; notify student. |

| Alternatives |  |
| --- | --- |
| 1a. Empty feedback where required. | 1a. Prompt to add comments. |
| 2a. No permission. | 2a. Deny with message. |

## Use Case 13: Manage Academic Terms

| Use Case | Manage Academic Terms |
| --- | --- |
| Actors | Coordinator |
| Pre-condition | Coordinator logged in |
| Post-Condition | Terms created/edited; single active term enforced |

| Flow of Events |  |
| --- | --- |
| Actor action | System Response |
| 1. Open Academic Terms. | 1. Show list and form. |
| 2. Add/edit term; set active. | 2. Save; enforce one active; confirm. |

## Use Case 14: Create Offerings and Enroll Students

| Use Case | Create Offerings and Enroll Students |
| --- | --- |
| Actors | Coordinator |
| Pre-condition | Active term exists |
| Post-Condition | Offering exists; students enrolled |

| Flow of Events |  |
| --- | --- |
| Actor action | System Response |
| 1. Create offering (course/section). | 1. Save offering. |
| 2. Enroll students (pick/upload). | 2. Attach students; confirm. |

## Use Case 15: Assign Faculty to Offerings

| Use Case | Assign Faculty to Offerings |
| --- | --- |
| Actors | Coordinator |
| Pre-condition | Offering exists; faculty available |
| Post-Condition | Faculty assigned; roles reflected |

| Flow of Events |  |
| --- | --- |
| Actor action | System Response |
| 1. Open offering; choose faculty. | 1. Validate and assign; update effective roles. |

## Use Case 16: View Group Progress

| Use Case | View Group Progress |
| --- | --- |
| Actors | Adviser |
| Pre-condition | Adviser assigned to group(s) |
| Post-Condition | Adviser sees current status and risks |

| Flow of Events |  |
| --- | --- |
| Actor action | System Response |
| 1. Open My Groups/Progress. | 1. Show KPIs and drill-down to milestones/submissions. |

## Use Case 17: View Calendar

| Use Case | View Calendar |
| --- | --- |
| Actors | All Users |
| Pre-condition | Logged in |
| Post-Condition | Events visible by role |

| Flow of Events |  |
| --- | --- |
| Actor action | System Response |
| 1. Open Calendar. | 1. Show defense schedules and events relevant to user. |

## Use Case 18: Manage Notifications

| Use Case | Manage Notifications |
| --- | --- |
| Actors | System (send); All Users (view) |
| Pre-condition | Triggering event occurs |
| Post-Condition | Notification delivered and can be marked read |

| Flow of Events |  |
| --- | --- |
| Actor action | System Response |
| 1. System detects event (invite, feedback, schedule). | 1. Create and queue notification. |
| 2. User opens notifications. | 2. List items; mark as read on view. |

If you need the document in Word, open this file and copy-paste the tables, or use a Markdown to DOCX/PDF converter (e.g., VS Code Markdown PDF extension, or pandoc).