

Operation contracts:

Contract CO1: Create a Task

Operation: createTask(title, description, priority, dueDate)

Preconditions:

- Title is not empty.
- Priority is a valid value.
- If dueDate is provided, check it is a valid date format.

Postconditions:

- A new Task instance t is created (object creation).
 - Attributes saved:---
 - t.title = title
 - t.description = description
 - t.priority = priority
 - t.dueDate = dueDate
 - t.creationDate is set to the current date/time
 - t.status = open
 - t is added/associated to the system's task collection
 - A new ActivityEntry instance a is created and associated with t, recording that the task was successfully created
-

Contract CO2: Mark a Task as Complete

Operation: completeTask(taskId)

Preconditions:

- A Task with an id= taskId exists.
- Task status is open (not already completed/cancelled).

Postconditions:

- Task t.status becomes Completed.
- A new ActivityEntry a is created and associated with t, recording that the task was completed (timestamped).
- a is associated with t (association formed) and records “completed task” with timestamp.

Failure scenarios:

- If task doesn't exist, the system reports error; no state change.
 - If task already completed/cancelled, then the system reports invalid action; no state change.
-

Contract CO3: Cancelling A Task

Operation: cancelTask(taskId)

Preconditions:

- Task with taskId exists.
- $t.status = \text{Open}$ (As it is not completed! Can't cancel a completed task...)

Postconditions:

- Task's $t.status$ becomes cancelled.
 - A new ActivityEntry is created and associated with t , recording the task's cancellation with timestamp.
-

Contract CO4: Assigning A Task To A Project

Operation: assignTaskToProject(taskId, projectId)

Preconditions:

- Task with taskId exists.
- Project with projectId exists.
- The Chosen Task is not a subtask (because subtasks cannot be assigned to projects directly)

Postconditions:

- Task t becomes associated with Project p (association formed).
 - If t was previously in another project, that association is removed.
 - A new ActivityEntry a is created + associated with t describing that the task was moved/assigned to a project p ... with a timestamp!
-

Contract CO5: Add A Tag To A Task

Operation: addTagToTask(taskId, tagName)

Preconditions:

- Task with taskId exists.
- tagName is not empty.
- The task does not already have a tag with name = tagName (avoid duplicates).

Postconditions:

- If a Tag instance g with g.name = tagName does not already exist yet, a new Tag object g is created (object creation).
- An association between t and g is formed
- A new ActivityEntry instance a is created and associated with t, recording that the tag tagName was added (and timestamped).

Failure scenarios:

- If task not found/doesn't exist, send an error, no state changed!
 - If tagName is empty, send an error, no state changed!
 - If tag already exists on task, send message + no action required!
-

Contract CO6: Updating A Task's details

Operation: updateTaskDetails(taskId: TaskId, title: String, description: String,
priority: Priority, dueDate: Date, status: Status)

Preconditions:

- Task with taskId exists.
- The task in question is not Cancelled

Postconditions:

-----The followings were successfully updated:-----

- t.title is updated to title
- t.description is updated to title
- t.priority is updated to title
- t.dueDate is updated to title
- t.description is updated to title
- t.priority is updated to title

-A new ActivityEntry instance a is created and associated with t, recording that the task was updated (timestamped too).