# Test Case (Task Scheduler with Priority reminder)

Prepared by : Julio Martinez

# API Detailed Test Cases

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| TEST CASE ID | TEST SCENARIO | TEST CASE | PRE-CONDITION | TEST STEPS | TEST DATA | EXPECTED RESULT | POST CONDITION | ACTUAL RESULT | STATUS (PASS / FAIL) |
| TC001 | Verify successful registration | Register - Success | API server is running and /api/register is accessible. | Send a POST request with a unique username and valid password. | { "username": "new\_user", "password": "pass123" } | { "success": true }, 200 OK | User is added to the database. | User is successfully created. | PASS |
| TC002 | Register with existing username | Register - Username Exists | User 'existing\_user' already exists in the database. | Send a POST request using the same username. | { "username": "existing\_user", "password": "pass123" } | { "error": "Username already exists" }, 409 | User is not added to the database. | Duplicate username error returned. | PASS |
| TC003 | Login with valid credentials | Login - Success | User account exists with correct credentials. | Send POST to /api/login with correct username and password. | { "username": "valid\_user", "password": "pass123" } | { "access\_token": "..." }, 200 OK | Access token is issued. | Access token returned. | PASS |
| TC004 | Login with invalid password | Login - Invalid Credentials | User account exists. | Send POST to /api/login with wrong password. | { "username": "valid\_user", "password": "wrong\_pass" } | { "error": "Invalid credentials" }, 401 | User is not logged in. | Error message returned. | PASS |
| TC005 | Logout and clear session | Logout - Clear Session | User is logged in with active session. | Send GET request to /api/logout. | Session with active username | Redirect to /index | Session cleared from server. | User redirected to index. | PASS |
| TC006 | Create a new task | Create Task | User is logged in and has permissions. | Send POST to /api/tasks with task details. | Task object with metadata | { "success": true }, 200 OK | Task is stored in the database. | Task created successfully. | PASS |
| TC007 | Fetch task list with auth | Get Tasks - With Authorization | User is logged in. | Send GET to /api/tasks. | None | Task list, 200 OK | List is displayed to user. | Tasks returned. | PASS |
| TC008 | Update task to mark as completed | Update Task - Mark as Completed | Task with ID 1 exists. | Send PATCH with completed=true. | { "completed": true } | { "success": true }, 200 OK | Task status updated in DB. | Task marked as completed. | PASS |
| TC009 | Delete existing task | Delete Task - Success | Task with ID 1 exists. | Send DELETE to /api/tasks/1. | None | { "success": true }, 200 OK | Task removed from DB. | Task deleted. | PASS |
| TC010 | Create a subtask under task 1 | Create Subtask - Success | Task with ID 1 exists. | Send POST with subtask title and status. | { "title": "Subtask A", "completed": false } | { "success": true }, 200 OK | Subtask added to task. | Subtask created. | PASS |
| TC011 | Mark a subtask as done | Update Subtask - Mark as Done | Subtask with ID 1 under Task 1 exists. | Send PATCH with completed=true. | { "completed": true } | { "success": true }, 200 OK | Subtask updated. | Subtask status updated. | PASS |
| TC012 | Delete a subtask | Delete Subtask - Success | Subtask exists under a valid task. | Send DELETE to /subtasks/1. | None | { "success": true }, 200 OK | Subtask deleted from DB. | Subtask deleted. | PASS |