API 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 | No user account exists with the provided username. | 1. Click register link 2. Enter credentials | "username": "new\_user", "password": "pass123" | User Registered Successfully | User is successfully created. | User is successfully created. | PASS |
| TC002 | Register with existing username | Register - Username Exists | User with the same username is already registered. | 1. Click register link 2. Enter same credentials | "username": "new\_user", "password": "pass123" | User already Exist | User is not added to the database. | Duplicate username error returned. | PASS |
| TC003 | Login with valid credentials | Login - Success | User account with matching username and password exists. | 1. Enter credentials 2. Click log in button | "username": "new\_user", "password": "pass123" | Log in Successfully and be redirected to dashboard | User session started and redirected to dashboard. | Logged successfully | PASS |
| TC004 | Login with invalid password | Login - Invalid Credentials | User account with matching username exists. | 1. Enter credentials with wrong password 2. Click log in button | "username": "new\_user", "password": "pass111" | Back to login page with error message | User session not started. Remains on login page. | Invalid credentials | PASS |
| TC005 | Logout and clear session | Logout - Clear Session | User is logged in with an active session. | Send GET request to /api/logout | Session with active username | Redirect to /index | User session cleared. Redirected to login page. | User redirected to index. | PASS |
| TC006 | Create a new task | Create Task | User is logged in and authenticated. | Send POST to /api/tasks with task details | Task object with metadata | { "success": true }, 200 OK | Task is created and stored in the database. | Task created successfully. | PASS |
| TC007 | Fetch task list with auth | Get Tasks - With Authorization | User is logged in and has existing tasks. | Send GET request to /api/tasks | None | Task list, 200 OK | User is able to view task list. | Tasks returned. | PASS |
| TC008 | Update task to mark as completed | Update Task - Mark as Completed | Task with specified ID exists and is not yet completed. | Send PATCH request with completed=true | { "completed": true } | { "success": true }, 200 OK | Task status updated in the database. | Task marked as completed. | PASS |
| TC009 | Delete existing task | Delete Task - Success | Task with specified ID exists in the database. | Send DELETE request to /api/tasks/1 | None | { "success": true }, 200 OK | Task is removed from the database. | Task deleted. | PASS |
| TC010 | Create a subtask under task 1 | Create Subtask - Success | Parent task (Task ID 1) exists in the database. | Send POST request with subtask title and status | { "title": "Subtask A", "completed": false } | { "success": true }, 200 OK | Subtask is created under the parent task. | Subtask created. | PASS |
| TC011 | Mark a subtask as done | Update Subtask - Mark as Done | Subtask under a valid task exists and is not completed. | Send PATCH request with completed=true | { "completed": true } | { "success": true }, 200 OK | Subtask status is updated in the database. | Subtask status updated. | PASS |
| TC012 | Delete a subtask | Delete Subtask - Success | Subtask exists under a valid task in the database. | Send DELETE request to /subtasks/1 | None | { "success": true }, 200 OK | Subtask is deleted from the database. | Subtask deleted. | PASS |