**Test Execution Run Document**

**1. Test Execution Summary**

* **Test Suite**: API Testing for Fakerest API
* **Total Test Cases**: 12
* **Tests Executed**: 12
* **Tests Passed**: 10
* **Tests Failed**: 2
* **Tests Blocked**: 0

| **Test ID** | **Test Case** | **Priority** | **Type** | **Status** | **Comments** |
| --- | --- | --- | --- | --- | --- |
| UT-001 | Validate GET /api/Books with valid data | High | Positive | **Passed** | Correct list of books returned. |
| UT-002 | Validate POST /api/Books with valid data | High | Positive | **Passed** | Book created successfully. |
| UT-003 | Validate GET /api/Books/{id} with invalid ID | High | Negative | **Passed** | Returned 404 error for invalid ID. |
| UT-004 | Validate DELETE /api/Books/{id} | Medium | Positive | **Passed** | Book deleted successfully. |
| UT-005 | Validate PUT /api/Books/{id} with empty payload | Medium | Negative | **Failed** | Returned 500 error instead of expected 400. |
| IT-001 | Create a book and verify it exists via GET /api/Books/{id} | High | Positive | **Passed** | Successfully fetched the created book. |
| IT-002 | Attempt to delete a book and verify via GET | Medium | Negative | **Passed** | Book no longer exists after deletion. |
| ST-001 | Validate all CRUD operations on Books | High | Positive | **Passed** | All operations worked as expected. |
| ST-002 | Test handling of invalid data across all endpoints | Medium | Negative | **Failed** | Some endpoints returned incorrect error codes. |
| E2E-001 | Create, update, fetch, and delete a book | High | Positive | **Passed** | All operations executed successfully in sequence. |
| PT-001 | Test GET /api/Books with 100 concurrent users | Medium | Load | **Passed** | Response time under 2 seconds for all requests. |
| PT-002 | Stress test POST /api/Books with increasing data | Low | Stress | **Passed** | System handled stress well, but degraded performance was observed at peak load. |

* **Test Execution Date**: [Enter Date]
* **Executed by**: [Your Name]

**2. Test Results Summary**

**3. Test Case Details and Logs**

* **UT-005: Validate**PUT /api/Books/{id}**with empty payload**
  + **Status**: Failed
  + **Error**: The API returned a 500 Internal Server Error instead of the expected 400 Bad Request for invalid input.

PUT /api/Books/1 Response Code: 500 Response Body: {"error": "Internal Server Error"}

* + **Next Steps**:
    - Investigate server-side validation for empty payloads.
    - Raise a defect for the incorrect error handling.
* **ST-002: Test handling of invalid data across all endpoints**
  + **Status**: Failed
  + **Error**: Several endpoints returned incorrect error codes (e.g., 200 OK instead of 400 for malformed data).

POST /api/Books Payload: {"title": "", "author": ""} Response Code: 200 Response Body: {"message": "Success"}

* + **Next Steps**:
    - Raise a defect for improper error handling.
    - Ensure proper validation for empty or invalid inputs.

**4. Conclusion and Next Steps**

* **Test Execution Summary**:  
  10 of the 12 test cases passed successfully. Two test cases failed, primarily due to error handling issues with the PUT and POST endpoints.
* **Next Steps**:
  1. Review and fix the error handling for invalid inputs on the PUT and POST endpoints.
  2. Re-test after fixes are applied.
  3. Continue with performance and integration testing for load/stress handling.

**5. Attachments**

* Screenshots of failed test case logs
* Test execution reports (Postman/JMeter output)