SESSION NOTES — ECSE 429 PART A

Build: runTodoManagerRestAPI-1.5.5.jar  
Area Under Test: Projects (incl. relationships & validation)  
Environment: macOS, localhost:4567, curl + jq  
Start (Date/Time): Oct 7–8 2025, ~01:00–01:45  #Duration: ~45 minutes  
Tester (Name / ID / Email): Suthiesan Subramaniam/ 261182188 / suthiesan.subramaniam@mail.mcgill.ca

CHARTER

Explore /projects endpoints focusing on:  
(a) CRUD (create, retrieve, update, delete) operations  
(b) task-linking relationships with /todos  
(c) error handling and validation behavior for missing/invalid inputs

Identify unexpected status codes or responses that deviate from REST semantics.

SESSION LOG

[01:00] Create Projects  
Requests:  
POST /projects {“title”:“S1 Project”,“completed”:false} → 201 Created (id = 5)  
POST /projects {“title”:“S2 Project”,“completed”:false} → 201 Created (id = 6)  
Observation: Creation works consistently with unique IDs assigned.

[01:05] List All Projects  
Request: GET /projects → 200 OK (list includes IDs 1, 3, 4, 5, 6)  
Observation: Older entries (“Office Work”, “BugHunt Project”) persist in database.

[01:10] Retrieve Specific Project  
Request: GET /projects/5 → 200 OK { “title”:“S1 Project” , “completed”:“false” }  
Observation: Returned in array wrapper (“projects”:[{…}]) → BUG-002.

[01:15] Update Project Fields  
Requests:  
PATCH /projects/5 {“title”:“S1 Project (Updated)”} → 200 (no JSON body)  
PATCH /projects/5 {“completed”:true} → 200 (no JSON body)  
Observation: PATCH succeeds but returns empty body → BUG-009.

[01:25] Create Todo and Link to Project  
Requests: POST /todos {“title”:“Linkable Todo”,“doneStatus”:false} → 201 Created (id = 23)  
POST /projects/5/tasks {“id”:“23”} → 201 Created  
GET /projects/5/tasks → shows todo id 23 linked under project 5.  
Observation: Task linking works and relationship persists → BUG-004 previously logged (duplicate link behavior also verified).

[01:30] Negative Tests  
GET /projects/999999999 → 404 Not Found {“errorMessages”:[“Could not find an instance with projects/999999999”]}  
POST /projects {} → 201 Created {“id”:“7”,“title”:““,”completed”:“false”}  
Observation: Empty payload incorrectly accepted → BUG-006 (validation gap).

[01:40] Deletion and Verification  
DELETE /projects/6 → 200 OK  
GET /projects/6 → 404 Not Found  
Observation: Deletion isolated to project only (no cascade) → BUG-007 not reproduced.

BUGS LOGGED / CLOSED

BUG-002: GET /projects/{id} returns array wrapper instead of single object.

BUG-004: Duplicate relationship link returns 201 Created (status mismatch).

BUG-006: Empty title accepted on project creation (validation missing).

BUG-009: PATCH returns empty body (no confirmation of update).

ISSUES / CONCERNS

Validation rules allow blank titles → data integrity risk (BUG-006).

Lack of PATCH confirmation breaks client update verification (BUG-009).

Wrapper on single GET adds unnecessary parsing complexity (BUG-002).

LIST OF TESTING IDEAS (NEW)

Re-test PATCH with Accept: application/xml to verify content-negotiation.

Try duplicate linking of same Todo → expect idempotent 201/409 behavior.

Send POST with very long title and observe field truncation/validation.

Simulate client retry sequence to evaluate duplicate creation behavior.

AREAS OF POTENTIAL RISK

Data inconsistency from blank titles or duplicate links.

Clients may misinterpret update success due to empty PATCH responses.

Wrapper in GET breaks standard JSON contract and may impact automation.

CAPABILITIES (EVIDENCE)

Scripts: ../CapabilitiesScripts/projectsCapabilities.sh + manual curl commands.

Terminal log: Oct 7–8 session (log attached).

Validated relations (Project↔Todo) through GET /projects/{id}/tasks.

SUMMARY OF FINDINGS

Project CRUD and linking features work for basic flows.  
PATCH operations return empty bodies (BUG-009).  
Validation is weak for missing title (BUG-006).  
GET response uses array wrapper (BUG-002).  
Duplicate linking behavior confirmed (BUG-004).  
Error handling correct for non-existent IDs (404).  
Overall, functionality is operational but response consistency and validation need improvement.