SESSION NOTES — ECSE 429 PART A

Build: runTodoManagerRestAPI-1.5.5.jar

Area Under Test: Todos (incl. relationships & media types)

Environment: macOS, localhost:4567, curl + jq

Start (Date/Time): Oct 5, 2025, ~22:35–23:20 #Duration: ~45 minutes

Tester (Name / ID / Email): Reswanth / 261108990 / Reswanth.rejipillai@mail.mcgill.ca

CHARTER

- Explore /todos behavior focusing on:

(a) relationships (todos↔projects, todos↔categories)

(b) media-type validation (JSON vs text/plain vs XML)

- Identify inconsistent status codes, response shapes, and type handling.

SESSION LOG

- [22:36] Setup new data

Requests: POST /projects {"title":"S1 Project"} → PID=4

POST /todos {"title":"S1 Todo","doneStatus":false} → TID=16

POST /categories {"title":"S1 Category"} → CID=11

Observations: All returned 201 Created.

- [22:36] Link todo to project (duplicate attempt)

Requests: POST /projects/4/tasks {"id":"16"} → 201

POST /projects/4/tasks {"id":"16"} → 201 (again)

GET /projects/4/tasks → shows one entry (id 16)

Observation: Duplicate link returns 201 but does not create a second link.

- [22:36] Unlink todo from project

Requests: DELETE /projects/4/tasks/16 → 200

DELETE /projects/4/tasks/16 → 404

Observation: Second unlink correctly reports 404.

- [22:37] Link category to todo (duplicate attempt)

Requests: POST /todos/16/categories {"id":"11"} → 201

POST /todos/16/categories {"id":"11"} → 201 (again)

GET /todos/16 → wrapper { "todos":[{ "categories":[{"id":"11"}] }] }

Observation: Same duplicate 201 behavior; single GET uses wrapper array.

- [22:38] Unlink category

Requests: DELETE /todos/16/categories/11 → 200

DELETE /todos/16/categories/11 → 404

DELETE /todos/16/categories/999999 → 404

- [22:38] Media types & validation

Requests: POST /todos (no Content-Type) → 415 Unsupported Media Type (good)

POST /todos (Content-Type: text/plain, JSON body) → 201 (id=17)

POST /todos {"title":" "} → 400 (validation: title empty)

POST /todos {"title":"wrong type","doneStatus":"notabool"} → 400

POST /todos (Content-Type: application/xml, XML body) → 201 (id=20)

Observations: text/plain accepted as JSON (201); XML input is accepted and returns JSON response.

- [22:39] Single GET & delete missing

Requests: GET /todos/16 → { "todos": [ { … } ] } (wrapper)

DELETE /todos/999999 → 404 Not Found

Observation: Wrapper confirmed; delete-missing returned correct 404 in this run.

BUGS LOGGED / CLOSED

- BUG-001 — doneStatus returned as string ("true"/"false") rather than boolean.

- BUG-002 — GET /todos/{id} returns array wrapper { "todos":[{…}] } instead of a single object.

- BUG-004 — Duplicate relationship link returns 201 Created (idempotency/status issue).

- BUG-005 — Content-Type: text/plain accepted for JSON body → 201 Created.

- (Note) BUG-003 (DELETE non-existent returns 2xx) was not reproduced in this session; keep open if seen elsewhere.

ISSUES / CONCERNS

- Status code semantics for duplicate links may mislead clients about creation vs idempotency.

- Media-type enforcement inconsistent (text/plain accepted as JSON).

- Boolean fields presented as strings force client special-casing.

LIST OF TESTING IDEAS (new)

- Try duplicate linking across other relationship endpoints.

- POST/PUT with Accept: application/xml and verify response content-negotiation.

- Boundary payload sizes (very long titles/descriptions).

- Concurrency/idempotency checks (repeat POSTs rapidly).

AREAS OF POTENTIAL RISK

- Client retries may cause false positives if server returns 201 on duplicates.

- Type mismatches (strings for booleans) may break strict clients.

- Wrapper on single GET may break client models expecting an object.

CAPABILITIES (EVIDENCE)

- Scripts used: ../CapabilitiesScripts/todosCapabilities.sh

- Ad-hoc curl commands listed above.

- Automated tests (Mocha/Chai) verified baseline behavior.

SUMMARY OF FINDINGS

- Duplicate relationship link returns 201 (no duplicate stored) → status mismatch (BUG-004).

- Media-type: text/plain accepted as JSON → 201 (BUG-005).

- Single GET uses wrapper array (BUG-002).

- Booleans returned as strings (BUG-001).

- Validation mostly correct: 415 on missing Content-Type, 400 on bad fields; 404 on delete-missing in this run.