Charter Introduction:

Identify capabilities and areas of potential instability of the "rest api todo list manager". Identify documented and undocumented "rest api todo list manager" capabilities.

For each capability create a script or small program to demonstrate the capability.

Exercise each capability identified with data typical to the intended use of the application.

Environment:

MacOS, Screen Resolution 1440x900 (Arturo Mory Ramirez)

MacOS, Screen Resolution 2560x1600 (Yida Pan)

Testers:

Arturo Mory Ramirez (260989669)

Yida Pan (260942327)

Date:

19:30 PM, 01/02/2024

Run:

java -jar runTodoManagerRestAPI-1.5.5.jar

Area:

Interoperability of the main functions and capabilities of the system

Session Breakdown:

45 minutes total

Test Detail:

Time	Tester	Request	Туре	Note	Output in JSON file
19:30	Arturo	http://localhost:456 7/projects	GET	Worked as expected, gets all the projects	

19:31	Arturo	http://localhost:456 7/categories	GET	Worked as expected, gets all the categories	Body Cookies Headers (4) Test Results
19:32	Arturo	http://localhost:456 7/categories/project s	GET	Worked as expected, gets all the projects in the categories (empty)	Dody Cookies Headers (4) Test Results Petty Raw Preview Visualize JSON V 55 1 {
19:33	Arturo	http://localhost:456 7/projects/categorie s	GET	Worked as expected, gets all the categories in the projects (empty)	Body Cookies Headers (4) Test Results Pretty Raw Preview Visualize 1 { 2 "categories": [] 3 }
19:35	Arturo	http://localhost:456 7/projects/1/categor ies	POST	Create relationship between a category and project with no category id, error message as expected	Body Cookies Headers (4) Test Results Pretty Raw Preview Visualize JSON 1
19:37	Arturo	http://localhost:456 7/projects/1/categor ies	POST	Create relationship with a proper id of "1"	Post Intgr/ficathost-4567/projects/ficategories Params Authorization Headers (8) 800/ * Per-request Script © none © form-data © x-www-form-unfercoded © raw © binary 1 (
19:39	Arturo	http://localhost:456 7/projects/1/categor ies	GET	Verify that for project id 1 a category has been created (successful relationship)	Body Cookles Headers (4) Test Results Pretty Raw Preview Visualize JSOI 1 {
19:40	Arturo	http://localhost:456 7/projects/categorie s	GET	Verify that for all the projects the category just created appears too	Body Cookies Headers (4) Test Results

19:42	Arturo	http://localhost:456 7/projects/1/categor ies/1	DELETE	Delete the relationship between the category and the project	DELETE Into://boahost-4587/projects//categories/1 Parama Authorization Headers (8) Body • Pre-request Script Osety Parama Key Volum Volum Body Cookles Headers (4) Test Results Petitly Raw Preview Visualize JSON > 1
19:44	Arturo	http://localhost:456 7/projects/1/categor ies	GET	Verify that it was deleted for projects and that categories is now empty	GET http://ocalhost-4567/projects/Ucalegories Params Authorization Headers (8) Body Pre-request So Query Params Key Value Noy Value Pretry Raw Preview Visualize JSON 1 { 2 "categozies": [] 3 } 3 }
19:45	Arturo	http://localhost:456 7/categories/1/proje cts	POST	Create new category of id 1 in project for category of id 1	Post http://locahost-4587/categories/1/projects Parama Authorization Headers (8) Body * Pre-request Scrig none © form-data © x-www-form-urbencoded © raw © bir 1 { 2
19:47	Arturo	http://localhost:456 7/categories/1/proje cts	GET	Verify that relationship has successfully been created	GET

19:49	Arturo	http://localhost:456 7/categories/1/proje cts/1	DELETE	Delete the relationship of project id 1 for category id 1	DELETE http://localhost-4567/categories/1/projects/1 Params Authorization Headers (8) Body Pre-request none form-data x-www-form-urlencoded raw 1
19:51	Arturo	http://localhost:456 7/categories/1/proje cts	GET	Verify that it was deleted for categories of id 1 and that projects is now empty	OET Intp://localhost.4567/categories/f/projects Params Authorization Headers (8) Body Pre-reque none form-data x-www-form-uriencoded raw 1
19:52	Arturo	http://localhost:456 7/categories/todos	GET	Verify that it was deleted for categories of all ids and that projects is now empty	Body Cookies Headers (4) Test Results Pretty Raw Preview Visualize Js 1 { 2 *todos*: [] 3 }
19:53	Arturo	http://localhost:456 7/todos/categories	GET	Retrieve successfully all categories for todos	Pretty Raw Preview Visualize JSON V 1
19:55	Arturo	http://localhost:456 7/categories/1/todo s	POST	Create a new todos of id 1 for category of id 1	Post V http://localhost.4567/categories/1/hodos Params Authorization Headers (8) Body Pre-rec none form-data x-www-form-urlencoded ras 1 { 2 ····*id*: *1* 3 } Body Cookies Headers (4) Test Results Pretty Raw Preview Visualize Json V
19:57	Arturo	http://localhost:456 7/categories/1/todo s	GET	Verify that the relation has successfully been created as now there is a todos for category 1	Body Cookies Headers (4) Test Results

19:59	Arturo	http://localhost:456 7/categories/1/todo s/1	DELETE	Delete the relationship of todos id 1 for category id 1	DELETE
20:01	Arturo	http://localhost:456 7/categories/todos	GET	Verify that it has been successfully deleted and that todos is now empty for all categories	Body Cookies Headers (4) T Pretty Raw Preview 1 { 2 "todos": [] 3 }
20:03	Arturo	http://localhost:456 7/todos	GET	Retrieve successfully all the todos	Pretty Raw Preview Visualize JSON
20:04	Arturo	http://localhost:456 7/todos/2/categorie s	GET	Retrieve as expected an empty categories for todos of id 2	Body Cookies Headers (4) Test Results Pretty Raw Preview Visualize JSON 1 {
20:06	Arturo	http://localhost:456 7/todos/2/categorie s	POST	Create a category with id 2 for the todos id 2	Post Intp://localhost-4567/todos/2/categories Parama Authorization Headers (8) Body Pre-request Sc. none form-data x-www-form-unercoded raw 1 1 2
20:08	Arturo	http://localhost:456 7/todos/2/categorie s	GET	Verify that the relation for todos id 2 and a category of id 2 has successfully been created	Body Cookies Headers (4) Test Results

20:10	Arturo	http://localhost:456 7/todos/2/categorie s/2	DELETE	Delete this category of id 2 for the todos of id 2	DELETE Intp://localhost-4567/lodex/2/categories/2 Params Authorization Headers (8) Body Pre-request none form-data *xwww-form-urlencoded raw { 1
20:12	Arturo	http://localhost:456 7/todos/2/categorie s	GET	Verify that the relation has been successfully deleted by retrieving a now empty categories for todos id of 2	Body Cookies Headers (4) Test Results Pretty Raw Preview Visualize JSON V 1 { categories*: [] 3 }
20:15	Arturo	http://localhost:456 7/todos/categories	GET	Verify that only the relation that we deleted has been successfully deleted, leaving with another already existing category of id 1 for todos of id 1	Body Cookies Headers (4) Test Results Pretty Raw Preview Visualize JSON 1 { 2