Class: System Integration

Exercise name: 04b Database Granular Access

Student name: Daniel Causevic

Student name that is being tested: Hero

Testing Documentation for Database Granular Access (SI_04b)

1. Test Setup

Database: PostgreSQL

Schema: accounts

Table: account

Users:

- Superadmin: Full read and write access.
- Read-Only User: Can read data but cannot modify it.
- No-Access User: No permissions to read or write data.
- Sue and Pepper: Read-only access to certain fields of their own data, with write access to their email addresses only.

2. Test Cases

2.1 Superadmin

Expected Behavior: The superadmin should have full access to the database, including the ability to view, insert, update, and delete any record.

Test Results:

Test 1: Full Select Access

The superadmin successfully retrieved all records from the account table, confirming full read access.

Test 2: Insert New Record

The superadmin was able to insert a new record into the account table, confirming write access.

Test 3: Update Existing Record

The superadmin successfully updated a record in the account table, confirming the ability to modify data.

Test 4: Delete Record

The superadmin was able to delete a record from the account table, confirming delete permissions.

2.2 Read-Only User

Expected Behavior: The read-only user should be able to view all records but cannot modify them.

Test Results:

Test 1: Full Select Access

The read-only user successfully retrieved all records from the account table, confirming read access.

Test 2: Attempt to Update

When attempting to update a record, the read-only user received a permission denied error, confirming that modification rights are properly restricted.

Test 3: Attempt to Insert

The read-only user received a permission denied error when attempting to insert a new record, confirming that insertion rights are properly restricted.

2.3 No-Access User

Expected Behavior: The no-access user should not be able to perform any database operations- not even reading.

Test Results:

Test 1: Attempt to Select

The no-access user received a permission denied error when attempting to view records from the account table, confirming no read permissions.

Test 2: Attempt to Insert

The no-access user received a permission denied error when attempting to insert a new record, confirming no write permissions.

Test 3: Attempt to Update

The no-access user received a permission denied error when attempting to update a record, confirming no modification rights.

2.4 Sue and Pepper (Restricted Access)

Expected Behavior: Sue and Pepper should have restricted access to their own records:

- They can read only certain fields (not `id` or `rolename`).
- They can update only their email address.

Test Results:

Test 1: Sue's Read Access

Sue was able to read only account_number, email, branch, and balance for her own record, confirming proper restricted access to certain fields.

Test 2: Sue's Restricted Access

Sue attempted to access all fields of the record, but the system correctly restricted access to `id` and `rolename` fields, confirming the enforcement of granular access policies.

Test 3: Sue's Update Access

Sue was able to update her email address successfully but could not modify any other fields. This confirms that Sue's update permissions are correctly restricted to her email field.

Test 4: Pepper's Read Access

Pepper was able to read only account_number, email, branch, and balance for his own record, confirming proper restricted access to certain fields.

Test 5: Pepper's Update Access

Pepper was able to update his email address successfully, confirming that Pepper's permissions are correctly limited to modifying his own email address.

3. Conclusion

Superadmin: Full access was confirmed, including insert, update, and delete operations.

Read-Only User: Successfully prevented from modifying data, but could view all records.

No-Access User: Confirmed that no actions were permitted (read/write).

Sue and Pepper: Successfully tested restricted access to certain fields and the ability to update only their email addresses.

All granular access control policies described in the documentation were effectively implemented and tested successfully. The database access control was configured properly at the cell and row levels, demonstrating fine-grained access control as intended.