Documentation

FinProjDevansh

Submitted To
Prof. Thomas BROUSSARD

Submitted By
M. Devansh D. SHARMA

SUBJECT DESCRIPTION

This project is aimed to provide a minimalistic application using which a user can perform all the CRUD operations.

In computer programming, **create, read, update, and delete** (as an acronym **CRUD**) are the four basic functions of persistent storage.[1]

Along with a few miscellaneous operations like fetching the list of identities in the database using the parameters specified. And terminating the program. The user is free to perform any and all of these operations any number of times as long as they don't conflict with the inner logic of the program, which will then raise exceptions, Guiding user to help modify the parameters he/she gave to the program so as to have a successful execution.

The Goal of this project has been to implement the knowledge acquired in the class, with many attempts of hit and trial and figuring out was wrong with the code, where i was just missing a semicolon.

Subject analysis

Features

This program offers the following the functionality.

Create Id : This function creates the ID of the user with the desired Inputs, like UID,
Delete Id : To delete an identity with an UID, Login Name, Email-Id.
Update Id : This updates the existing IDs in the database, The program cannot update any identities which are not existing in the system, So the user cannot try to fool JAVA, But the user is still welcomed to try to do so. Be warned, JAVA is smart!
User Verification/Authentication : This is the actual verification by comparing the user input data from the console to the data store in the database, and if they match, the user is allowed access to the further functionalities of the program i.e the CRUD operations.
Fetch the list of IDs: This feature fetches the list of IDs stored in the database by using UID, Email-ID, Login Name.

Application Feasibility

The application is minimalistic and just contains the core features so as to not distract the user from what he expects from this program and to put the performance of the program first over features that are only supplemental.

The user can provide their preffered choice of Login name, password and email from the console with ease and can create their account. Now if the User is not satisfied with his preferences given earlier and wants to update some parameters or even delete his ID, he/she can do so easily.
The user is also guided with exceptions to help him/her to correct their responses to an acceptable input.

Data Description

The JDBC connection works as a middleman between the program and the derby database. And all of the user stores data is stores in the Derby Database.
Using the basic queries like Inset and create, update and delete the users and the IDs are maintained which is very important.

Status | Result1

	UID	LOGINNAME	EMAIL_ID
1	1	log	log@log.log
2	2	biscotti	bis@bis.bis
3	4	juillet	jui@jui.jui

Fig 1: Table of IDs

Status | Result1

	LOGINNAME	PASSWORD
1	ds@epita.fr	dsepita
2	sk@epita.fr	skepita

Fig 2: Users table

Expected Results

Authentication/Verification: User can only access the CRUD operations after clearing this step successfully. The data they enter should match the data in the database.
After successful Id verification, The user is free to perform the following CRUD operations

1. Create Id
2. Delete Id
3. Fetch list of all Ids Id(Non Crud OP)
4. Update Id
5. Stop and Exit.(Non CRUD OP)

Scope of Application - Limits

The main downside of this project is that is unsecure and does not have GUI. GUI are essential in today's world and the program is practically useless without a gui for a common user who does not have a derby database installed in his machine. This program is probably less secure than a potato. No password encryption, No user information encryption, no user management, basic Id management,No User and Identity relation, The need to install an additional database to store all the user input data. All of these things make this program a very basic project, that is good enough for learning the basics of java but in this project is not something that would be useful in the real world.

Scope of Application - Evolutions

The program can benefit hugely with just three things

- Security, Encryption of personal data(Thanks to GDPR).
- User Interface.
- Me enhancing my java skills.

Conception

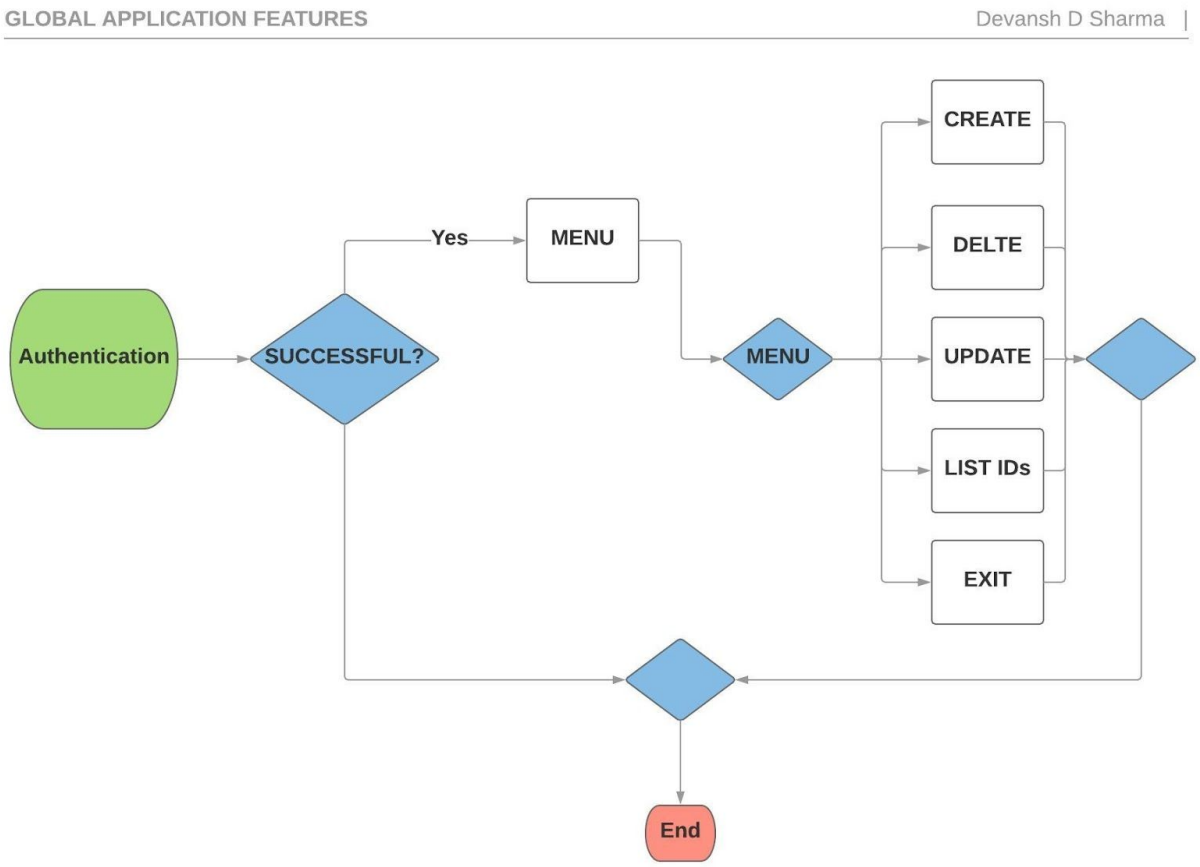
Chosen algorithm

For executing the CRUD operations multiple times, While loop is used with a console based application.

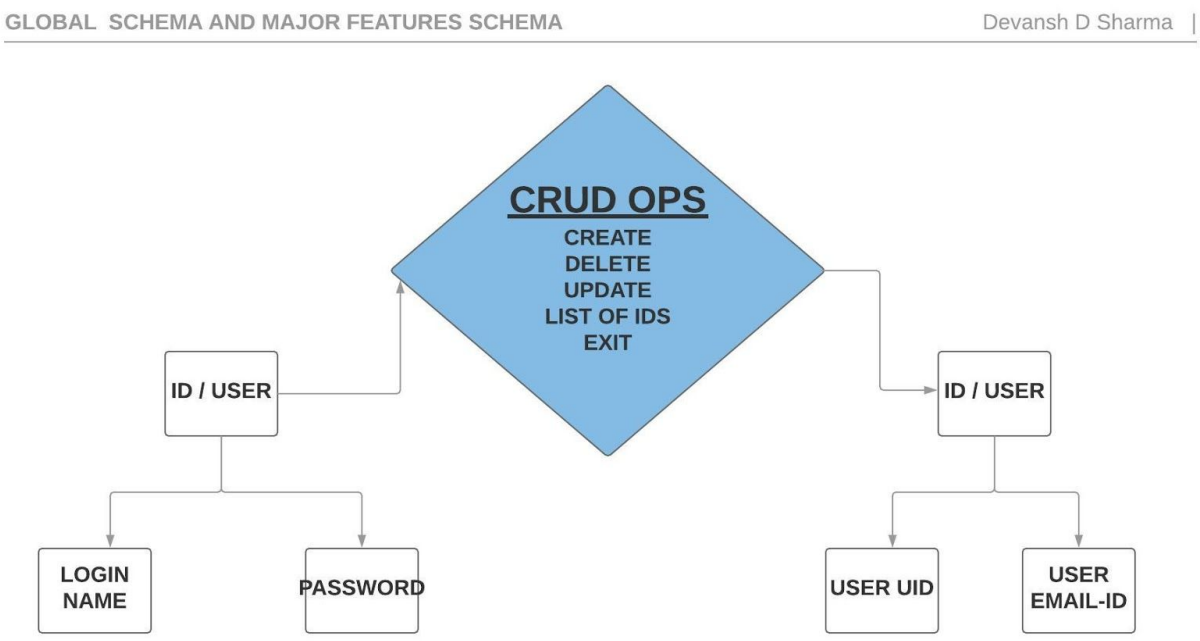
Data structures

Lists and Strings were used widely in the project for storing the identities.

Global Application Flow



Global schema and major features schema



Console operations description

Authentication/Verification:

This is the first step that the user must complete in order to proceed further and access the crud operations. The authentication is done by comparing the values stores in the derby database and the values entered by the user through the console.
If the values match the user will be permitted to go to the next step and can continue executing the crud operations an otherwise he/she has to retry or exit the program.
The ID_JDBC_DAO will return TRUE or FALSE based on the parameters entered by the user. If the Authentication and verification with the Derby Database is successful then the ID_JDBC_DAO will return TRUE otherwise it will return FALSE. But the user is free to try again.

Create ID:
After the use has successfully logged in. The create ID option is open to them and they can create any ID. With their desired specifications It requires the User to enter UIC, Login name, password and an email-id
This data is then stored in the derby database in the Identities table.

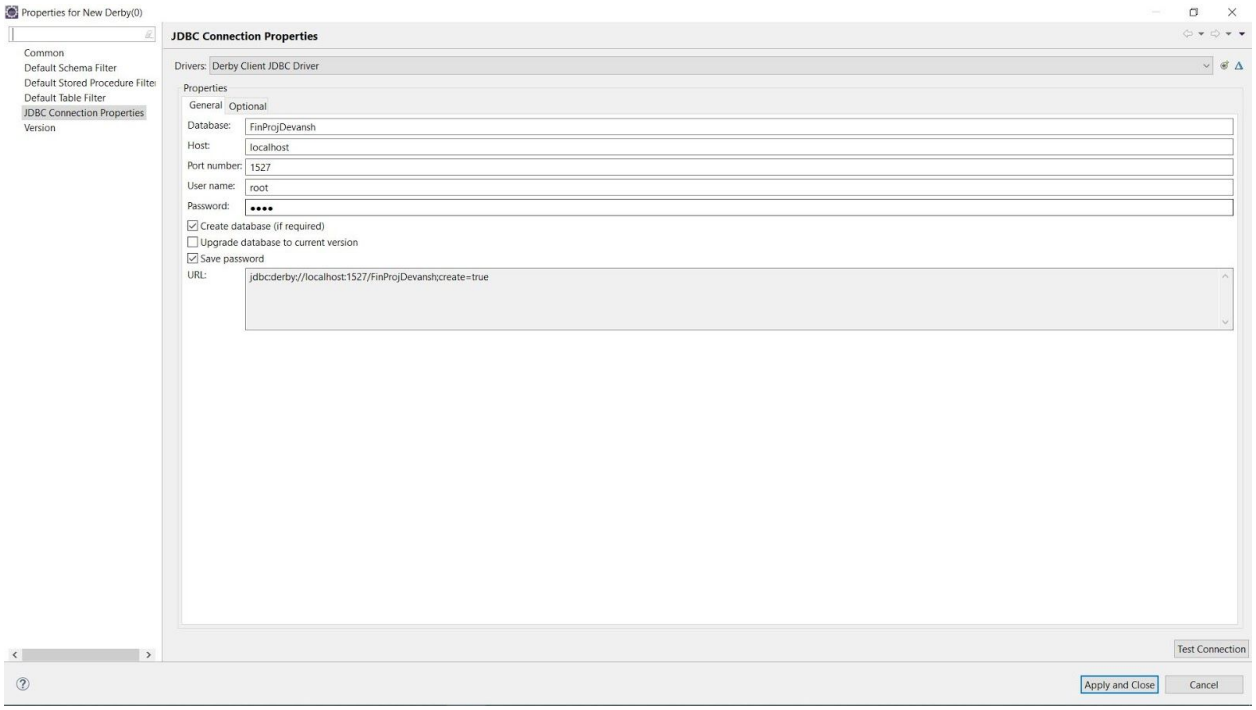
Delete ID:
After the use has successfully logged in. The delete ID option is open to them and they can delete any ID. Only the UID is required to delete an ID. And the user can not execute this function if the ID they are trying to delete does not exist.
This deletes the entry from the Identities table in the database.

Update ID:
After the use has successfully logged in. The update ID option is open to them and they can Update any ID. Only the UID is required to execute this operation and they can enter the new values that they want to choose.. And the user can not execute this function if the ID they are trying to update does not exist.
This overwrite the previous value in the identities database

List ID: This function gives the list of IDs that are stored in the database. If the user wants to fetch this list they need only the UID

Exit:
This feature stops the execution of the program.

Configuration Instructions



References

1 Wikipedia.com
2 Lectures and Class notes!
3 http://www.thomas-broussard.fr
4 Quora.com
5 Stackoverflow
6 W3 schools
7 Youtube tutorials.