Access Control - DBMS for ThisIsSpartanTech Application

Access Control is a vital component in maintaining the integrity of the backend of the “ThisIsSpartanTech” application. The database must have roles and privileges in place that ensure that the following actions are available for each role:

*Student:*

* Students can access their own current information (Select all from RegisteredFor table) but *cannot* access this information for other students.
* Students can add or drop courses belonging to them, but *not* to other students (Insert/Delete on CRN and StudID in RegisteredFor table).

*Professor:*

* Professors can view grades for courses that a student is registered for taught by that professor (Select Grade from RegisteredFor).
* Professors can change students’ grades and course description (Modify Course table - Description, Update RegisteredFor - Grade) for courses they teach but *not* the student grades and course description for courses they do not teach.

*School Administrator:*

* School administrators can access and modify (Insert/Delete/Update) student, professor, and course information (all from Student, Professor, and Course tables respectively) for departments they are administrators for but cannot access student and professor information for departments they are *not* administrators for.

*DBA:*

* Database Administrator has access to all privileges for all tables in the DBMS and the ability to grant the above-mentioned privileges to Student, Professor, and School Administrator roles.

Roles for this DBMS include DBA (Database Administrator), Student, Professor, and School (Department) Administrator.

These are the only roles to be assigned within the system. Classification of users based on login information is handled by code of the application, rather than the database.

SQL statements for granting the above-mentioned privileges and roles is as follows:

*DBA:*

## 

## CREATE ROLE DBA IDENTIFIED BY PASSWORD;

## 

## GRANT SELECT ON STUDENT, DEPARTMENT, PROFESSOR, SCHOOLADMIN, COURSE, REGISTEREDFOR, STUDENTRECORD, WORKSFOR TO DBA;

## 

## GRANT INSERT, DELETE, UPDATE ON STUDENT, DEPARTMENT, PROFESSOR, SCHOOLADMIN, COURSE, REGISTEREDFOR, STUDENTRECORD, WORKSFOR TO DBA;

*School Administrator:*

CREATE ROLE SADM IDENTIFIED BY PASSWORD;

GRANT SELECT ON STUDENT, PROFESSOR, COURSE, STUDENTRECORD, REGISTEREDFOR TO SADM;

GRANT INSERT, DELETE, UPDATE ON STUDENT, PROFESSOR, REGISTEREDFOR TO SADM;

GRANT INSERT, UPDATE ON COURSE(COURSENUM, SECTIONNUM, COURSENAME, DESC, SEMESTER, YEAR) TO SADM;

*Professor:*

CREATE ROLE PROF IDENTIFIED BY PASSWORD;

GRANT UPDATE ON REGISTEREDFOR(GRADE) TO PROF;

GRANT SELECT ON REGISTEREDFOR (STUDID, CRN) TO PROF;

GRANT SELECT ON STUDENTRECORD(STUDID, COURSENUM, GRADE, LETTERGRADE, SEMESTER, YEAR) TO PROF;

GRANT SELECT ON COURSE(COURSENUM, SECTIONNUM, PROFID) TO PROF;

GRANT INSERT ON COURSE(DEPTKEY, COURSENUM, SECTIONNUM, PROFID, PROFNAME, DESC, SEMESTER, YEAR) TO PROF;

GRANT INSERT, DELETE, UPDATE ON COURSE(DEPTKEY, COURSENUM, SECTIONNUM, PROFID, PROFNAME, DESC, SEMESTER, YEAR) TO PROF;

GRANT DELETE ON COURSE TO PROF;

*Student:*

CREATE ROLE STUD IDENTIFIED BY PASSWORD;

GRANT INSERT ON RegisteredFor(STUDID, CRN) TO STUD;

GRANT DELETE ON RegisteredFor(STUDID, CRN) TO STUD;

GRANT SELECT ON COURSE(CRN, DEPTKEY, SECTIONNUM, PROFNAME, DESC) TO STUD;

GRANT SELECT ON REGISTEREDFOR(COURSENUM, GRADE) TO STUD;

CREATE VIEW STUDENTGRADE AS

SELECT (COURSENUM, GRADE) FROM REGISTREDFOR WHERE STUDID = :OLD.STUDID;

GRANT SELECT ON STUDENTGRADE TO STUD;