DBS Project 8

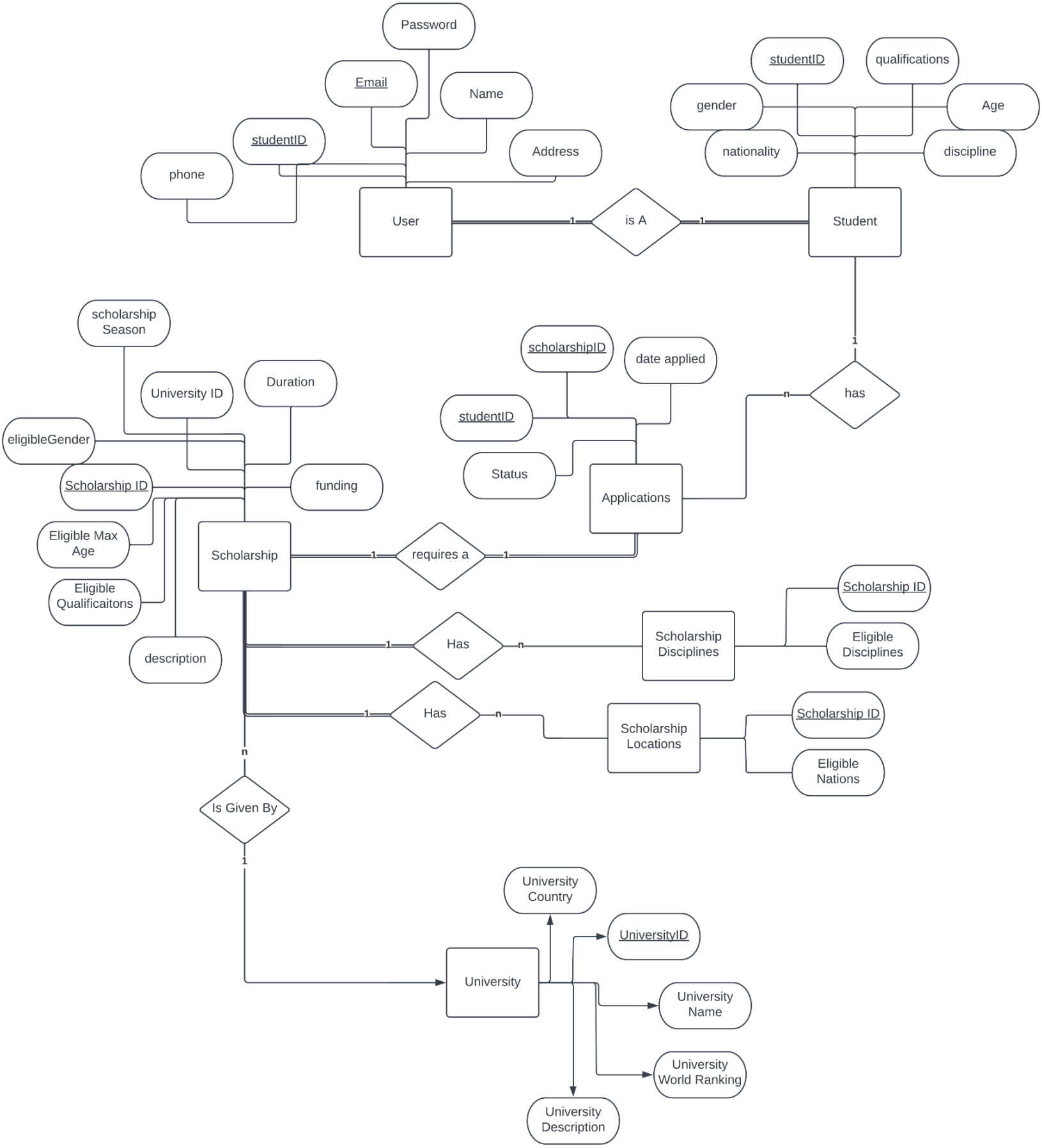
Scholarship Help System Documentation

Group 86

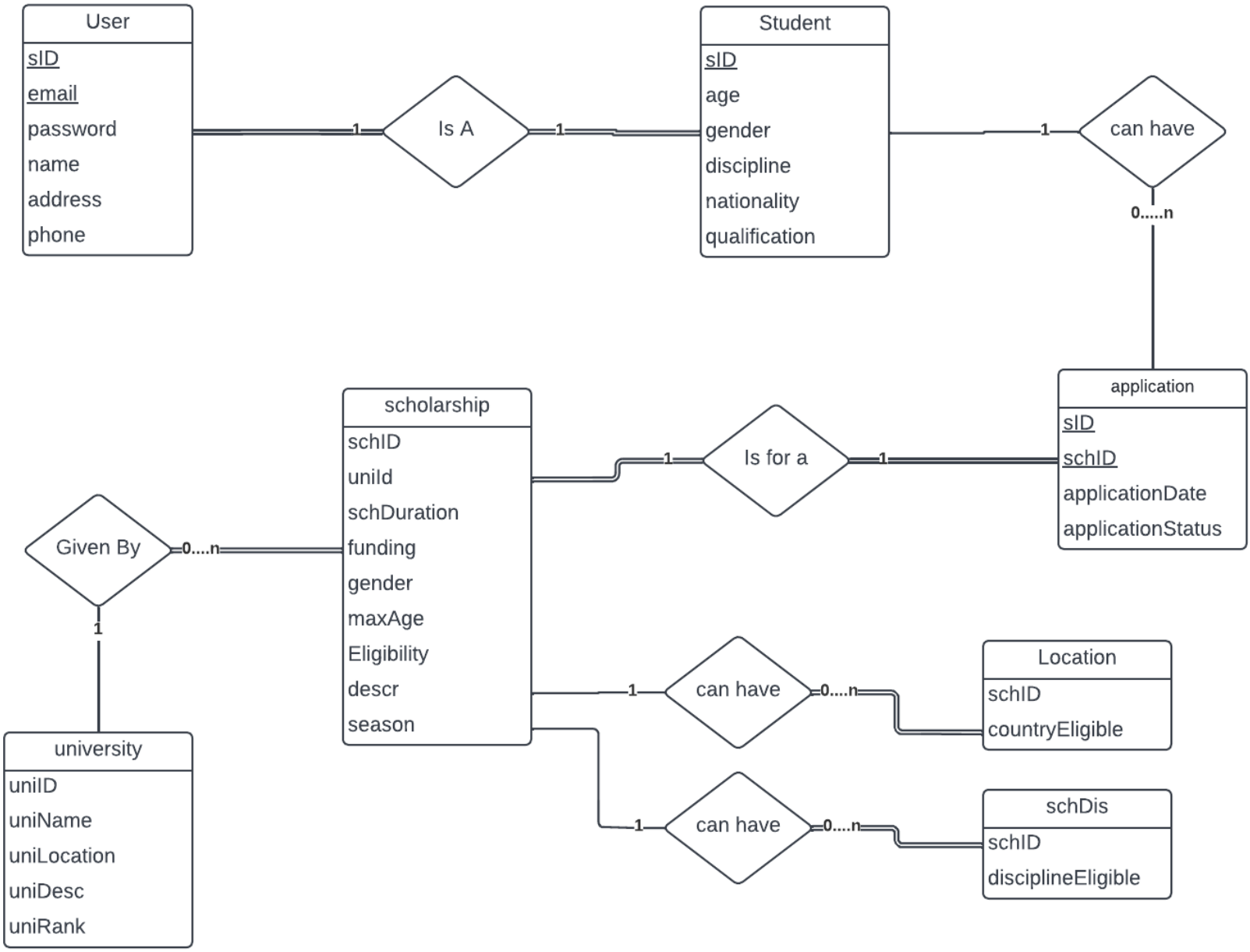
Sushrut Patwardhan 2019B5A70382P

Tanishq Duhan 2019B5A70636P

Entity Relation Diagram



Schema Diagram



Data Normalization

All the data is in tabular form and hence atomicity is observed implying data is in 1 NF form.
The functional dependencies are as follows:

User(sid, email, password, name, address, phone)

Sid -> email, password, name, address, phone

University(uniID, uniName, uniLocation, uniDesc, uniRank)

uniID -> uniName, uniLocation, uniDesc, uniRank

Scholarship(schID, uniID, schDuration, funding, Gender, maxAge, season)
schID -> uniID, schDuration, funding, Gender, maxAge, season

ScholarshipDisciplineEligile(schID, discipline)
schID-> discipline
ScholarshipCountryEligile(schID, country)
schID-> country

Application(sID, schID , applicationDate, applicationStatus)
(sID, schID) -> applicationDate, applicationStatus

Student(sID, age, gender, nationality,discipline, qualification)
sID -> age, gender, nationality,discipline, qualification

User(sID, email, password, name, address, phone)
(sID, email) -> password, name, address, phone

As we can see all the functional dependencies do not contain partial dependencies or transitive dependencies, we can clearly say it is in 2NF as well as 3 NF form.
All the Multi-Values Dependencies are in separate tables such as a particular scholarship can be available for multiple countries as well as for multiple disciplines, for this we have 2 different tables thus making the database in 4NF form.

List Of Tables Required

For the proper working of the scholarship portal, the following tables need to be present:

1. **User**: This table contains the email ID, the password for authentication, and other details of the user.
2. **Student**: This table contains the unique ID of the user, and contains other information that is required for checking the eligibility of the students for scholarships like the qualifications of the student, his/her age, gender, discipline and nationality of the student.
3. **Application:** This table maps the student with the scholarships he/she has applied for. This contains the student ID, the scholarship ID, the date of application and the status(rejected, pending and approved) of the application.
4. **Scholarship**: This table contains the details and eligibility of various scholarships with the ID of the university that offers the scholarship. It contains the scholarship ID, University ID, the duration of the scholarship, the funding offered, the gender it is applicable for, the max-age it is applicable for, the qualifications it requires the student to have, the description of the scholarship and the season(spring, summer, winter and fall) the scholarship is offered for.
5. **Location:** This is an optional table that contains additional restrictions on the scholarship. It contains the Scholarship ID and the countries whose students can apply for the scholarship.
6. **SchDis**: This is an optional table that contains additional restrictions on the scholarship. It contains the Scholarship ID and the disciplines whose students can apply for the scholarship.
7. **University**: This table contains the details of the universities. One university can give one or more scholarships but one scholarship can only be given by a particular university. This table contains the university ID, name, country of location, description of the university and the world ranking of the universities.

In our project, we have generated and used data(not necessarily real) for all tables.We have included 10 Users and 10 corresponding Students, 2 Applications, 500 Scholarships, 1000 Locations, 1000 SchDis and 500 Universities.

Additional components

We use 8 major procedures for updating and querying tables:

1. **userCheck**: this procedure takes the email ID and password and authenticates the user. If the details are found to be correct, it then calls the Login procedure.
2. **Login:** this procedure takes the unique sID associated with each user's email ID and returns the student details of the user. This data is then used to display the applications the user has already registered for and the scholarships the student can apply for using other procedures.
3. **Register**: in the case of a new user, he/she has to register first to use the scholarship portal. The new user has to provide his/her- name, email address, password, address, phone number, age, gender, discipline, qualification and nationality. The email is first checked if already exists using another checkMail procedure. If the email is new, the user details are added to the database and the user can now check for applicable scholarships and apply for them.
4. **checkMail:** this procedure takes the entered email ID and checks if it already exists in the database. If it does, the registration is denied.
5. **getAppliedScholarships:** this procedure takes in the studentId and returns the details of the scholarships the student has applied for.
6. **scholarshipQuery:** this procedure takes in the studentID and returns all the scholarships that the student is eligible for. If the student wishes, he/she can apply for the scholarship using the addApplication procedure.
7. **addApplication**: this procedure takes in the studentID and scholarshipID, and registers the student for the scholarship (add the student ID, scholarship ID, the current date and the default application status “pending” in the application table).
8. **getUniversity:** this procedure takes in the universityID and returns the details of the university.