

The Relational Schema of our EERD:

User (ID (PK), Name, Phone No., Email, Address)

//the ID is a unique and not null which make us use it as a primary key for the user entity

Student (Student ID (FK referencing User ID), CGPA, Application State, Major, Academic Level, Tech Skills, Certification, LinkedIn Profile, Uni. Mentor ID (FK referencing ID of the Uni. Mentor))

//we used the user ID as a foreign key cause each Student ID is actually a User ID

//we used the Uni. Mentor ID as a foreign key cause each uni. mentor is mentoring many students

Mentor (Mentor ID (FK referencing User ID), Years of Experience, Department, Job Title, Office Hrs., Max Assigned Students)

//since each mentor ID is a user ID, we also used the user ID as a foreign key

//since User have a disjoint relation with Student and mentor, we used Method A of schema which explains why both student and Mentor has the user ID as their foreign key

Uni. Mentor (Uni. Mentor ID (FK referencing Mentor ID), Uni. Name, Feedback Score, Assigned Level, Years of Experience, Department, Job Title)

//the university mentor take the mentor ID as a foreign key

Company Mentor (Company Mentor ID (FK referencing Mentor ID), Company Name, Assigned Branch, Evaluation Feedback, Company Logo (FK referencing to the company))

//the Company mentor take the mentor ID as a foreign key

//Since the company mentor work for a certain company it takes the primary key of the company entity as a foreign key

Company (Company Logo (PK), Name, Industry, Website, Student ID (FK referencing Student ID))

// we choose the company logo as the primary key for the company as it can be unique and not Null

//the company takes the student ID as a foreign key cause a student have a one-to-many relation with the company as each student can apply for many companies

Company Location (Company Logo (FK referencing to the company), Locations)

//Since the company may have many locations, we choose to make it multi-valued attribute in the company so, as shown we represent the location in another table with company logo which is the primary key of the company as its foreign key

Internship Application (Application ID (PK), Company Mentor ID (FK), Uni Mentor ID (FK), Student ID (FK), Status, Applied Date, Decision Date, Doc. ID (FK))

//Application ID is the unique attribute which made it the primary key

//as the company mentor, the university mentor, the student and the Academic docs. Have a one-to-many relation with the internship application we take the primary key from each entity and make it the foreign key of internship application

Has a Relation (Company Logo (FK reference to the Company), Application ID (FK reference to the Internship Application))

//since the internship application have a many to many relations with the company as many internship applications are sent to many companies, we represented the relation between both in another table having both the primary key of the internship application and the primary key of the company as its foreign keys

Evaluation Report (Report ID(PK), Evaluation_ Date, Company Mentor ID(FK), Performance_ Score, Feedback)

//the report ID is unique so chosen to be the primary key of the Evaluation report

//as each company mentor can submit many evaluation report then it have a one to many relation with the evaluation report so we take the primary key of the company mentor as a foreign key in the evaluation report

Academic Docs (Doc ID(PK), Uploaded_ By, Uni Mentor ID (FK), Timestamp, Transcript, Recommendation Letter)

//the Doc. ID is unique chosen to be the primary key of the Academic Docs.

//as each university mentor can upload many academic Docs. Then its one-to-many relation so we take the primary key of the uni. mentor as a foreign key in the academic docs.

Normalization in 3NF:

User Table

Attributes: User_ ID, Name, Phone Number, Email, Address

- **1NF:** ✓ All fields are atomic
- **2NF:** ✓ All non-key attributes fully depend on User_ ID
- **3NF:** ✓ No transitive dependency

User Table is in 3NF

Student Table

Attributes: Student_ ID, CGPA, Application State, Major, Academic Level, Tech Skills, Certification, LinkedIn Profile, User_ ID

- **1NF:** ✓ Assuming Tech Skills and Certification are stored in atomic form (like one skill per row)
- **2NF:** ✓ Student_ ID is PK; all attributes fully dependent
- **3NF:** ✓ No transitive dependencies; User_ ID links to another table

Student Table is in 3NF

Mentor Table

Attributes: Mentor_ ID, Years of Experience, Department, Job Title, Office Hours, Max Assigned Students, User_ ID

- **1NF:** ✓ Atomic fields
- **2NF:** ✓ All attributes fully dependent on Mentor_ ID
- **3NF:** ✓ No transitive dependencies

Mentor Table is in 3NF

University Mentor Table

Attributes: Uni_ Mentor_ ID, Uni Name, Feedback Score, Assigned Level, Mentor_ ID

- **1NF:** ✓ All atomic
- **2NF:** ✓ Fully dependent on Uni_ Mentor_ ID
- **3NF:** ✓ Mentor_ ID is FK, and no derived attributes

University Mentor Table is in 3NF

Company Table

Attributes: Company_ Logo, Company Name, Industry, Website

- **1NF:** ✓ Atomic and unique
- **2NF:** ✓ All attributes depend on Company_ Logo
- **3NF:** ✓ No non-key dependencies

Company Table is in 3NF

Company Mentor Table

Attributes: Company_ Mentor_ ID, Company Name, Assigned Branch, Evaluation Feedback, Mentor_ ID, Company_ Logo

- **1NF:** ✓ Atomic
- **2NF:** ✗ Company Name depends on Company_ Logo, not on Company_ Mentor_ ID

Decomposition: Split into two:

1. **Company_ Mentor Table:** Company_ Mentor_ ID, Assigned Branch, Evaluation Feedback, Mentor_ ID, Company_ Logo
2. **Company Table:** Company_ Logo, Company Name, Industry, Website

After this split, both tables meet 3NF

Fixed: Now in 3NF

Company Location

Composite Key: (Company_ Logo, Location)

- **1NF:** ✓ Atomic
- **2NF:** ✓ Fully dependent on full composite key
- **3NF:** ✓ No transitive dependency

Company Location Table is in 3NF

Internship Application

Attributes: Application_ ID, Status, Applied Date, Decision Date, Company_ Mentor_ ID, Uni_ Mentor_ ID, Student_ ID, Doc_ ID

- **1NF:** ✓ Atomic
- **2NF:** ✓ Single key (Application_ ID), fully dependent attributes
- **3NF:** ✓ No transitive dependencies

Internship Application Table is in 3NF

Application_ Company_ Mentor (Join Table for many-to-many)

Attributes: (Application_ ID, Company_ Mentor_ ID)

- **1NF:** ✓ Composite key, atomic
- **2NF:** ✓ Fully dependent
- **3NF:** ✓ No transitive dependency

Application_ Company_ Mentor is in 3NF

Has_ Relation Table

Composite Key: (Company_ Logo, Application_ ID)

- **1NF:** ✓ Atomic
- **2NF:** ✓ Fully dependent
- **3NF:** ✓ No non-key attribute to be transitively dependent

Has_ Relation is in 3NF

Evaluation Report

Attributes: Report_ ID, Evaluation Date, Company_ Mentor_ ID, Feedback

- **1NF:** ✓ Atomic fields
- **2NF:** ✓ Report_ ID as PK, all dependent
- **3NF:** ✓ No derived/transitive dependency

Evaluation Report is in 3NF

Performance Score

Attributes: (Report_ ID, Student_ ID) → Score

- **1NF:** ✓ Composite key used correctly
- **2NF:** ✓ Score depends on full key
- **3NF:** ✓ No derived dependency

Performance Score is in 3NF

Academic Docs

Attributes: Doc_ ID, Uploaded_ By, Timestamp, File Type, Uni_ Mentor_ ID

- **1NF:** ✓ All values atomic
- **2NF:** ✓ Doc_ ID is PK, full dependency
- **3NF:** ✓ No transitive dependency

Academic Docs Table is in 3NF