# Name of Your Company

A&H Group

# Project Title

**Student Clubs Management Database System**

# Team

* Ali Abbasimoghaddasi (MySQL Expert)
* Harsheen Kaur Cheema (MS SQL Server Expert)

# Weekly Meeting Hours

We will meet and work on the project every Wednesday from 9 pm.

# Project Description

* Each club has a Lead.
* Each club can have several Groups.
* Each group has a Head.
* Students can only join these groups.
* Members of group can be students currently registered or graduated.
* One Student can be part of many groups of a club or many groups.
* Each group organizes some Events.
* Each event is organized by a group, date, time, location (room number, floor, building), subject, and registration fee.
* We store the information about organizers of an event.
* For alumnus, we need to store his/her work history information (name of company, position, start date, end date…).
* A person may join a company in different time. person may join a company, leave it, and join in again several years later.
* We store start date and end date of membership in each group that a student join.
* Groups may work on some Funded Projects.
* We store how much money (fund or budget) a project receives.

# Assumptions about Cardinality and Participations

* A club is led by one student. (Total participation on club side, 1:1)
* A group is headed by one student. (Total participation on group side, 1:1)
* Student joins groups. (Total participation on group side, M:N)
* A person joins a company in different times. (Recursive relationship)
* Groups work on funded projects. (Total participation on projects side, M:N)
* Groups organize events. (Total participation on events side, M:N)

Screenshots of the applications installed:

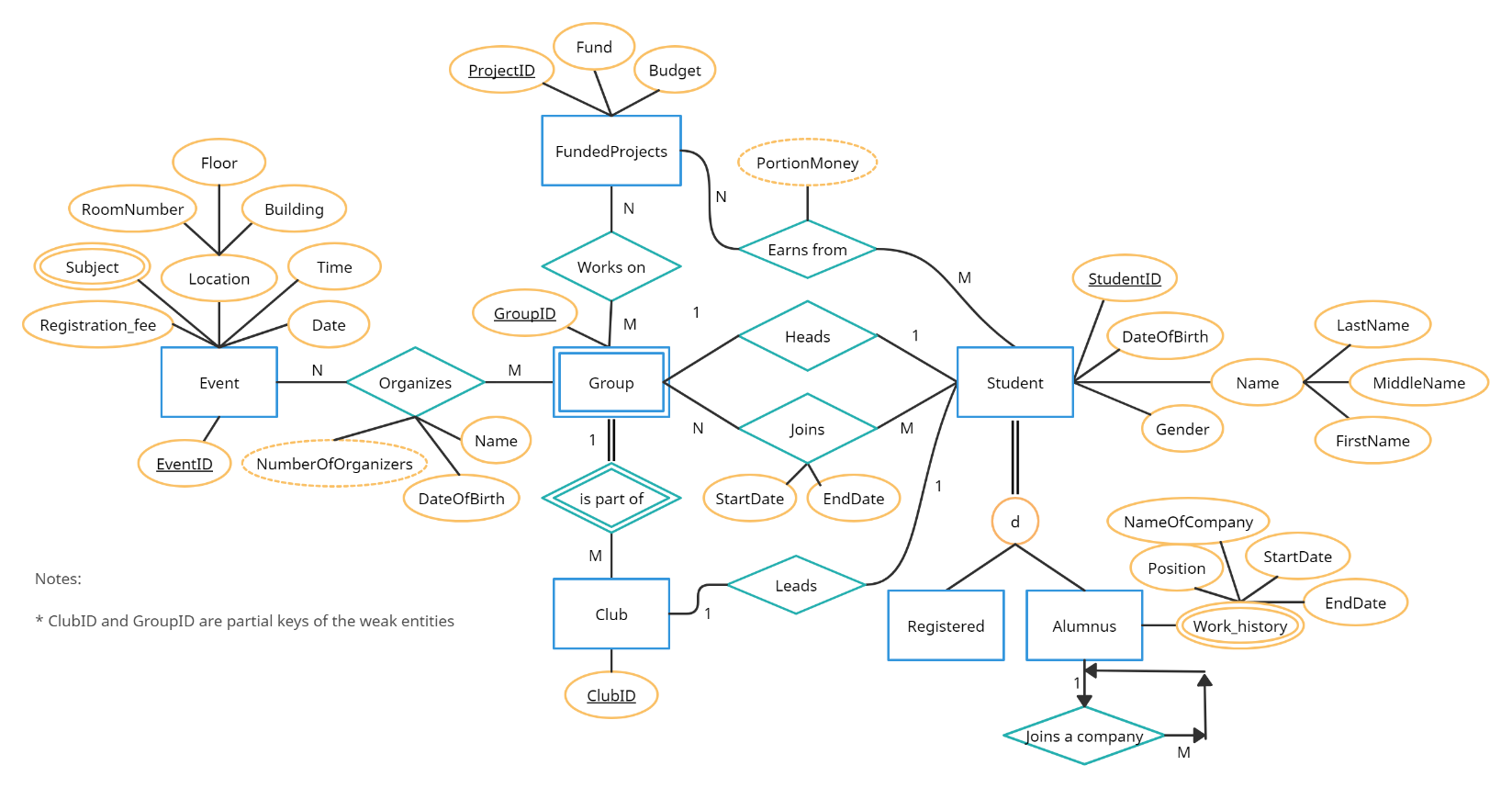
Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

# EER Modeling Diagram



# ER-Model Mapping to Database Relational Schema

Event (EventID, Registration\_fee, Date, Time, RoomNumber, Floor, Building)

FundedProjects (ProjectID, Fund, Budget)

Club (ClubID)

Student (StudentID, DateOfBirth, Gender, LastName, MiddleName, FirstName)

Register (**StudentID**)

Alumnus (**StudentID**, Position, NameOfCompany, StartDate, EndDate)

Event\_Group (**EventID**, GroupID, Name, DateOfBirth)

Club\_Group (**ClubID**, GroupID)

FundedProject\_Group (**ProjectID**, GroupID)

Student\_Leads\_Club (**StudentID, ClubID)**

Student\_EarnsFrom\_Funded (**StudentID, ProjectID**)

Group\_Student (**GroupID, StudentID**, StartDate, EndDate)

Event\_Subject (**EventID**, Subject)

# Normalization

our functional dependencies are as below:

EventID → Registration\_fee, Date, Time, RoomNumber, Floor, Building, DateOfBirth, Name, GroupID

ProjectID → Fund, Budget, StudentID, GroupID

ClubID → StudentID, GroupID

StudentID → DateOfBirth, Gender, LastName, MiddleName, FirstName, ProjectID, GroupID, ClubID, Position, NameOfCompany,

StartDate, EndDate

GroupID → Name, DateOfBirth, EventID, ProjectID, ClubID, StudentID, StartDate, EndDate

**Our candidate keys are:**

EventID

ProjectID

ClubID

StudentID

GroupID

1NF: To achieve 1NF we will have to deal with the two multi-valued attributes that are [Subject] and [Work\_History]. In order to do that, from relational mapping, we have separated the multi-valued attributes from our table so that they don’t need to be normalized anymore.

2NF: First condition is applied because it is already in 1NF. The second condition is also applied since there is no partial dependency.

Therefore, the table is already in 2NF.

3NF: First condition is applied because it is already in 2NF. The second condition is also applied since there is no transitive dependency. Therefore, the table is already in 3NF.

BCNF: First condition is applied because it is already in 3NF. The second condition is also applied since the left hand side is a super key. Therefore, the table is already in BCNF.

# Determining Data Types (Domain) and Constraints

All the ( EventID, ProjectID, ClubID, StudentID, GroupID, Registration fee, RoomNumber, FloorNumber, BuildingNumber, Fund, Budget ) attributes are set as INT data type because there’s only positive digits included in each of them.

Attributes ( Gender, LastName, MiddleName, FirstName, Subject ) are set as a CHAR datatype because they are all treated as string objects, and we only need to deal with characters in these cases. And the reason we didn’t use VARCHAR datatype for them is that VARCHAR would slow the database which would make a less efficient datatype.

Other Constraints used in our database are Date and Time which are used for mapping our Event.

# Creating and Dropping Database and Tables - SQL DDL

File Name: CreateTable.sql

File Name: DropTable.sql

# Inserting Values in Tables

File Name: Insertion.sql

# SQL Queries

File Name: Students\_SqlQuerries.sql