

# UHD-ACM User Database

---

## 1. Team members:

- Ruiz, Brian
- Zubair, Eashrak

## 2. Project Abstract

As part of our University Database Management Systems (CS4318) course we were designated to develop a small-scale relational database project to explore --in a practical sense-- the distinct phases of database system development process. Which includes Design, SQL Implementation, and Management thereof the respective system. Given that members of our team were involved and participated with leading a university student club, ACM, and seeing that there was a growing amount of participants for the club itself, the decision was made to create a database which supports CRUD operations would aid the management of the students accounts for the club.

## 3. Mission Statement

The primary purpose of the UHD ACM database is to maintain the data generated about current and potential members while being able to supply information that supports the engagement services and activities provided to members based on their membership status, and the management services necessary by the clubs administrators.

## 4. Mission Objectives

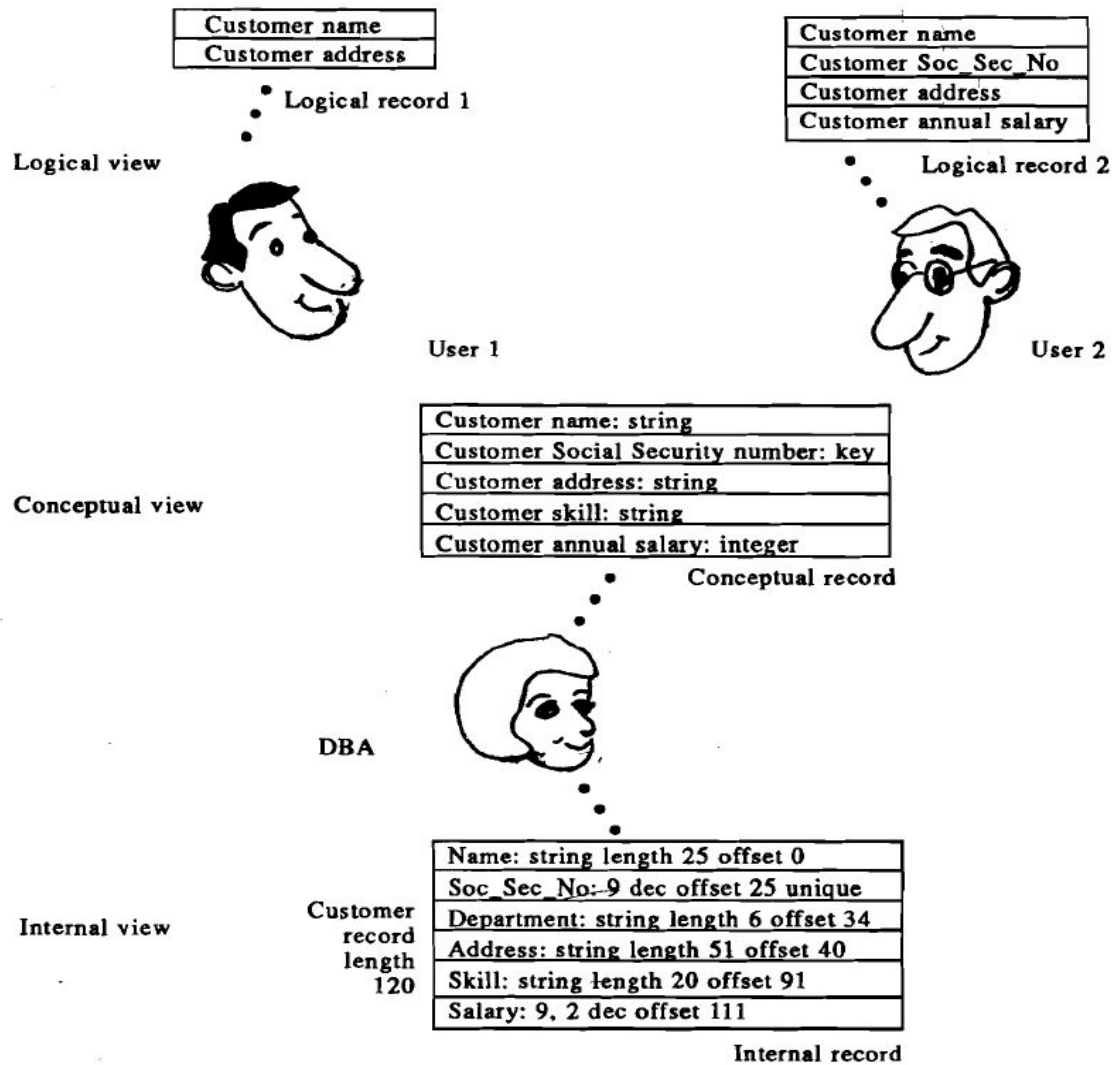
A chief objective of the database system is designing it with high consideration for scalability, as we witness that ACM club is experiencing an almost exponential growth in student interests, and thus should be adaptable to any upcoming changes with respect the clubs needs. All whilst acting as the medium to collect and store newly incoming data in terms of attributes and tuples, and thus grouped into distinct relations. Lastly to maintain consistency among the operational data to ensure levels of integrity.

## 5. Database Constraints

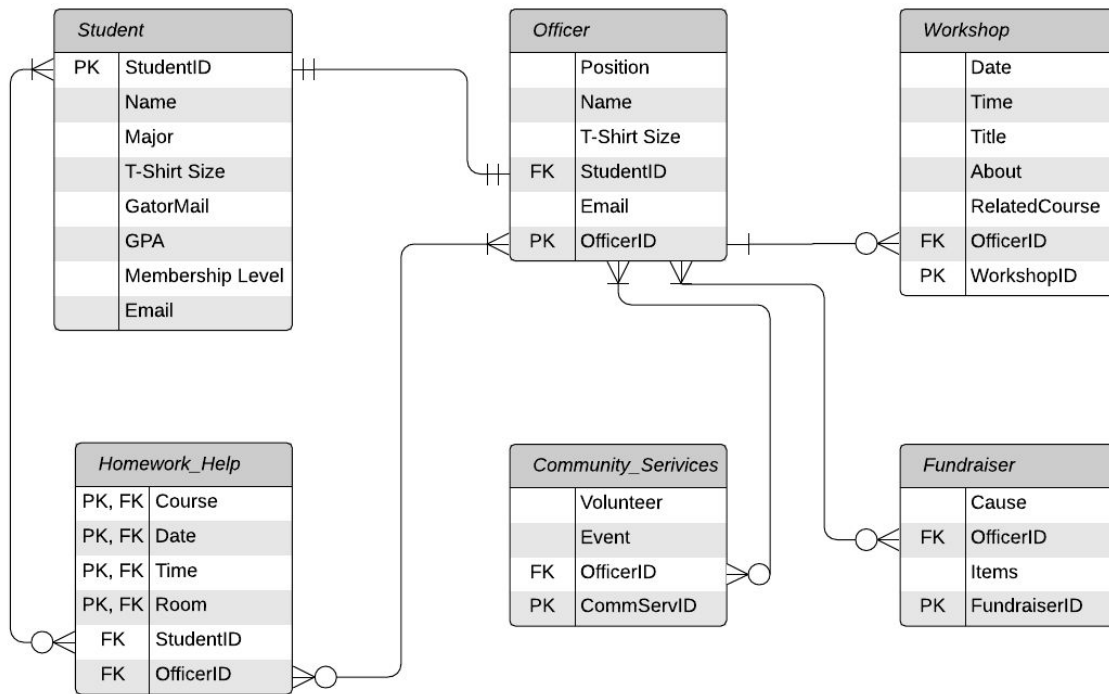
In order for data to protect itself, constraints on variables of data must be established in order for our database to maintain its validity. Formatting in Standard Query Language..

- *Student/OfficerID* = **PRIMARY KEY** (combination of not-null & unique)
- *WorkshopID/FundraiserID/CommServeID* = **NOT NULL**
- *GatorMail* **CHECK** for (sql) substring domain "@gator.uhd.edu" or "@uhd.edu"
- *Student/OfficerID* = **FOREIGN KEY**
- *Workshop.Title* **DEFAULT** = *WorkShopID* (If no workshop title, default to workshopID)

## 6. Major User Views



## 7. E/R Diagram (underlined key attributes for entity sets, relationships)



## 8. Use Cases

Use case name: **Joining ACM**

Actor: **Student (Insertion)**

Steps:

- 1) User clicks the "Join ACM" button
  - a) A new tuple is generated for the Student ID
- 2) User is asked to enter Student ID, Name, Major, T-Shirt Size, GatorMail, GPA, Membership Level, and Email address
- 3) All information is displayed and the user is prompted to confirm if the information is correct
- 4) User clicks confirm button

Use case name: **Joining ACM**

Actor: **Officer (Update)**

Steps:

- 1) User clicks the "Join ACM" button

- 2) User is asked to enter Student ID, Name, Position, T-Shirt Size, and Email address
- 3) The Student ID is added into the Officer ID, which will be used to identify the officer from henceforth
- 4) All information is displayed and the user is prompted to confirm if the information is correct
- 5) User clicks confirm button

Use case name: **Leaving ACM**  
Actor: **Student (Deletion)**  
Steps:

- 1) User clicks the “Leave ACM” button
- 2) The tuple containing that user’s Student ID is deleted from the Database

Use case name: **Asking for Homework Help**  
Actor: **Student**  
Steps:

- 1) User clicks the “Get HW Help” button
- 2) A HW Help ID is generated using the Course, Date, Time, and Room. For example, A student asking for help with Web Programming on May 5th 2019 at 3pm in room N420 will generate a HW Help ID called WebProgramming\_050519\_3p\_N420

Use case name: **Add Community Services**  
Actor: **Officer**  
Steps:

- 1) User clicks the “Add Community Service” button
  - a) A CommServID is created
- 2) User is asked to enter the Officer ID of the officer monitoring the event, the name of Event, and if students can Volunteer

Use case name: **Delete Community Services**  
Actor: **Officer**  
Steps:

- 1) User clicks the “Delete Community Service” button
- 2) The tuple containing the CommServID is deleted from the Database

Use case name: **Create Fundraisers**  
Actor: **Officer**

Steps:

- 1) User clicks the “Add Fundraiser” button
  - a) A FundraiserID is created
- 2) User is asked to enter the Officer ID of the officer monitoring the fundraiser, the Cause of the fundraiser, and the Items required for that fundraiser

Use case name: **Delete Fundraisers**

Actor: **Officer**

Steps:

- 3) User clicks the “Delete Fundraiser” button
- 4) The tuple containing the FundraiserID is deleted from the Database

Use case name: **Create Workshops**

Actor: **Officer**

Steps:

- 1) User clicks the “Add Workshop” button
  - a) A WorkshopID is created
- 2) User is asked to enter the Officer ID of the officer hosting the workshop, the Title of the workshop, a little description About the workshop, Date and Time of when the workshop will be hosted, and the academic course that is closely related to this workshop

Use case name: **Delete Workshops**

Actor: **Officer**

Steps:

- 5) User clicks the “Delete Workshop” button
- 6) The tuple containing the WorkshopID is deleted from the Database