

# Context - Registration Data

## Deliverable

A logical ERD design for the registration process' relational database. This includes table names, field names, and labels indicating which fields are primary vs foreign keys. Relationships between tables should be shown using the Crow's Foot method.

---

## Background

The University of Texas at Austin has struggled to manage its student and registration data in a consistent, uniform manner. As part of a UT-wide start-up challenge, we are being tasked to design a simple proof of concept database that could be used as a model for a new system to manage student, faculty, and class registration. Since this is a proof of concept it will only manage to store the more critical data that is needed in the system and will not need to include every possible feature. It just needs to store courses, room assignments, teachers, student information, and section enrollments.

---

## Student record creation

When a student is set up in the system we need to track their general info like name, age, contact information, student classification, and address. For address, each student will have two addresses and only two (i.e. *home address* which is usually their address prior to moving to Austin, and their *local address* which is where they live during the school year). Note that a student can be tied to one home and one local address but an address can be shared among many students. We don't anticipate needing to store more than these two addresses for each student but we're not sure about this.

---

## Course Creation process

Despite what we've seen in the past where two teachers could teach a course, it's rare. So, to keep things simple a single section will be tied to only one teacher. A teacher can teach many sections. A course is broken into one or many sections, but a section can only tie back to one specific course. A classroom is assigned to a specific section for a specific day and time. In other words, one section will be held in one room during a

specific day and time, but a room can tie to many different sections and times in a single semester.

---

## Registration process

When a student signs up for a course, they can only sign up for one specific section of that course. A student will have many sections from courses they will enroll in, and a section will have many students. Each student should be given a single grade or pass for the section they're tied to or a Q or W if they dropped or withdrew.

---

## Technical Details

**Teachers** – Every teacher should have a first and last name. We don't need to track middle names of teachers. Teachers will have one primary department code that we tie them to but they are allowed to teach sections of courses outside of their department. Teachers can have many titles especially if they are part-time on staff and part-time on faculty, but for the purposes of this system, we will just track their primary teaching title (e.g. Professor). Each faculty will have an office number and phone number. Each faculty member will also have a full address. We don't assume they're all in Austin or Texas so they can have unique cities, states, and zip codes. Many addresses require a second address line for situations like a Dept/Suite number or more info than a single line address can manage. Also, we need to store a single campus\_mail\_code for each faculty member (e.g. B6500)

---

**Students** – For students, we require a first and last name. If they have a middle name we'll want to capture it but it's not required. We will need to track a student's date of birth, and classification of whether they're a freshman, sophomore, junior, senior, grad student, Ph.D., or continuing ed. Student classification will be stored as a simple code. We also need to know their primary contact information (email and phone) and their primary college code. Some students will be assigned a college code like Business (B) or Engineering (E) but if not, they'll have a default of Undergraduate Studies (i.e. U). We also will need to store a flag on the student to identify if they're an international student or not. Every student will eventually declare a major and they're allowed to declare more than one. A major will have its own code (e.g. MIS) and description (Management Information Systems). We also want to store the cumulative GPA of each student and while this could be derived, it will speed up decision-making when we need to have the GPA easily accessible.

---

**Course Sections** – A course will have a course code (e.g. MIS325), course name (e.g. Intro to Database Mgmt), and course description which is a longer description that shows up in the course catalog. A room will have a building\_code that identifies what building it's in. We'll also want to track the floor it's on, the room's official room number, and the max number of seats in the room. While a section is tied to a specific room, a room can be assigned to many sections since there are many time slots in a day that a room is available for. Each section will be assigned a number from 10005 to 99995. The number doesn't uniquely identify a specific section because we reuse the numbers, but the combination of a number and a semester\_code will create the unique combination to identify a specific section. A section will occur on a specific day or set of days during a specific hour (2 - 3:30pm). A section has a specific mode (e.g. Face to face, Online, Hybrid) and a seat limit that can be less than the room's max seats but no greater. Each section will have a status to indicate if the section is open, closed, or some other status. We'd also like to have quick access to the number of students currently enrolled in a section. Even though this could be derived, we think it will aid in query writing and be more efficient.