This University registration model gives the brief preview of the student schema, related information and their relationships controlled by the Administrator

**1. Register new student:**

University receives the application from the people(users) who want to study here. While applying for the University, he/she provides the following info.

     user({

Includes all the attributes from this field that includes {username: , pswd: ,fName: , lName: ,email: ,pp: ,address: ,highSchool: ,country: ,graduateYear: , interestedMajor: ,Bday: ,Phone: ,someText or queries

})

After approved by the university, a unique ID is assigned to each user for accessing in future.

**2. Add Student to Collection:**

Let’s say the collection includes, the collection of students who are registered each year(could be the primary key). There will be students from graduate or undergraduate level with different interested majors or undecided. So Administrator will add the student info in following basis:

year

…..

….

2015

2016{

  Graduate: {},

  Undecided: {},

  Undergraduate: {

  Major: {

  ID: { Reference to the unique user ID}

              Username: , everything from the user information. }

   }

}

     }

**3. Registration for Classes**

Student registers for the classes. Now Administrator will store the following information

A. CourseSchema({

this includes all the courses that students registers and courses includes title: , author: , courseID: ,, courseProf: ,and every information.

})

A. SemesterSchema({

this includes all the attributes from this field that includes {year: eg, freshman, junior.., author:{includes student username and unique ID}}, whole courses(referencing to courseSchema)

})

**4. Update Student Information:**

After admitting to University and registering required courses, the administrator will update the user information that includes attributes from the user model. We will make assumptions that the user includes not only attributes from the “Register new student” page but also the courseSchema which can be included by referencing to all models that you want to include. But this time, as the courses are registered, new information about the financial report will be added to that userID and get updated. At this point, the administrator will put a hold on that user and will set an automated email to be sent asking to pay ASAP.

**5. Check the financial Status of student:**

Ofcourse, you have to pay for all the courses you registered. As unique ID is given to each user in the beginning of the registration process. Each time, Administrator will check the “hold” list and can access to each student through that unique ID.

If (amount is already paid)

      remove the hold.

**update();**

Else{

    if(date Crossed)

Remove from the list.

**DropOut();**

   Else

Set the email and scare them.

       }

**6. Edit or Update Profile:**

All the recent and updated information.