

## **CSC-540-project Milestone 2**

### **Team Members:**

**Gaurav Aradhye (garadhy)**

**Aditya Shirode (avshirod)**

**Anuja Supekar (asupeka)**

**Kahan Prabhu (kprabhu)**

### **Application Constraints:**

We identified following application constraints (which can't be handled through the database alone).

1. The number of credits the student is taking for a course should be less than or equal to the maximum credits specified for the course AND greater than or equal to minimum credits specified for the course.
2. Student must enroll in the course which has participation level matching to the participation level of the student.
3. Student enrolling in the course should conform to some/all prerequisite criteria.
4. Student should be allowed to enroll in the course as long as there are seats available for a particular course offering, and later should be allowed to enroll in the waitlist associated with the course offering as long as waitlist limit is not reached.
5. Student should be allowed to enroll in the course only before the course addDate.
6. Student should be allowed to drop the course only before the course dropDate.
7. WaitlistNumber of students will be decremented accordingly in case students with lesser waitlistNumbers dropping or getting enrolled into the course.
8. Student should not get enrolled into a course in case there is already a course registered for the student with an overlapping time slot.

## Functional Dependencies:

### Table: Unity Users

unityID  $\rightarrow$  fname  
unityID  $\rightarrow$  lname  
unityID  $\rightarrow$  password  
unityID  $\rightarrow$  email  
unityID  $\rightarrow$  gender  
unityID  $\rightarrow$  dateOfBirth  
unityID  $\rightarrow$  address  
unityID  $\rightarrow$  phone  
unityID  $\rightarrow$  userType

### Table: Employee

unityID  $\rightarrow$  SSN

### Table: Faculty

unityID  $\rightarrow$  deptID  
unityID  $\rightarrow$  Field  
SSN  $\rightarrow$  deptID  
SSN  $\rightarrow$  Field

### Table: Department

deptID  $\rightarrow$  deptName

### Table: Course

courseID  $\rightarrow$  sectionID  
courseID  $\rightarrow$  title  
courseID  $\rightarrow$  maxCredits  
courseID  $\rightarrow$  minCredits  
courseID  $\rightarrow$  participationLevel

### Table: Students

unityID  $\rightarrow$  deptID  
unityID  $\rightarrow$  overallGPA  
unityID  $\rightarrow$  residencyLevel  
unityID  $\rightarrow$  participationLevel  
unityID  $\rightarrow$  yearEnrolled

**Table: Billing**

unityID, billID → billAmount

unityID, billID → dueDate

unityID, billID → outstandingAmount

**Table: CreditRequirements**

We have decided to use participationLevel and residencyLevel as composite primary key as against using the creditReqID.

participationLevel, residencyLevel → maxCreditsReq

participationLevel, residencyLevel → minCreditsReq

**Table: BillingRate**

participationLevel, residencyLevel, semesterID → \$/CreditHour

**Table: Semester**

semesterID → season

semesterID → year

semesterID → addDate

semesterID → dropDate

semesterID → semStartDate

semesterID → semEndDate

**Table: CourseOffering**

courseID, sectionID, semesterID → classDays

courseID, sectionID, semesterID → startTime

courseID, sectionID, semesterID → endTime

courseID, sectionID, semesterID → location

courseID, sectionID, semesterID → totalSeatsOffered

courseID, sectionID, semesterID → seatsAvailable

courseID, sectionID, semesterID → waitlistLimit

**Table: SpecialPermissions**

specialPermissionID → specialPermissionCode

**Table: Prerequisite**

currentCourseID corresponds to courseID in CourseOffering table.

prereqCourseID corresponds to courseID in Course table.

currentCourseID, sectionID, semesterID, prereqCourseID → grade

**Table: Transcript**

unityID, courseID, sectionID, semesterID → grade

unityID, courseID, sectionID, semesterID → numberOfCredits

**Table: Waitlist**

unityID, courseID, sectionID, semesterID → waitlistPosition