The DB Conceptual Design Document (CDD): CSRR Database

Submitted By:

Naga Sai Prithvi Raj Mamidala (101834072)
Sai Aditya Thalluri (101863023)

INDEX

1. The Miniworld (or Universe of Discourse)

Miniworld for the project is Computer Science Department. A Computer science department is part of university organization. It deals with the students registering in that specific department. It provides the data about students, employee, staff etc. belong to that department. Data Base is being built for keeping record of information about student grades, courses, major courses required to graduate and employee dealing with specific courses, staff's role, their wages etc.

The purpose of the current project is to design and build a database to support the efforts of student recruitment and retention and to track students' progress. This database build to have record of their employees, students, programmes they offer, requirements to get graduated for a specific program. Tracking the information like grade of students, semester they are in, number of courses completed, prerequisites of courses, courses to take to get graduated, PhD students projects they are working on, their advisor, research grants etc. Internship opportunities provided by any professor, organization of social activities to promote student networking, recruitment activities etc.

Users for this database are employees, staff, department students. Department students check the core courses they require to get graduated. Students, can have track of student's participation in student organizations, record of student's jobs in summer, their internship opportunity etc. Department can have record of their students working, the list of courses per any specific semester and faculty teaching that course.

2. Conceptual Schema of the Database

Notations and Definitions

The notation used: all upper case for the entity names, lower case for the relationship names, and the first letter capitalized for attribute names.

The description of the entities starts with a sentence which explains their meaning. Then the attributes to describe the instances are included. The relationships are also described by a sentence and a list of attributes if it has them.

Each attribute has a four-letter code which describes the type of attribute according to the four classification criteria for attributes.
The format for this code is: (xyzw), where
x tells that the attribute is simple (S) or composite (C),

y tells that the attribute has a single value (S) or is multivalued (M), z tells that the attribute is primitive (stored) (P) or derived (D), in case it is derived, an explanation of how to deduce it from other attributes or a formula must be specified, and

w tells that the attribute is fixed (F) (i. e. it must have a value that is not null) or optional (O).

2.1 Entities

The entities for this database are:

- ACTIVITY
- COURSE
- EMPLOYEE
- FACULTY
- INCIDENT
- JOB
- PROGRAM
- RESEARCH_PROJECT
- STUDENT
- STAFF
- SURVEY
- TUTOR
- ON_CAMPUS
- INTERNSHIP

A detailed description of each entity follows.

ACTIVITY: It is an entity, organization of social activities that students participate for social networking.

Some of the example instances are: Sigma chi, gamma, Aurora.

Attributes:	**ActivityName**	(SSPF)
	ActivityType	(SSPF)
	OrganizedBy	(SSPF)
	ContactInfo(EMail,Phone)	(CSPO)

COURSE: Courses is an entity a student enrols for completing program opted or to gain knowledge.

Example: DBMS, AI, Data mining

Attributes:	**CourseID**	(SSPF)
	CourseName	(SSPF)
	CreditHours	(SSPF)
	Department	(SSPF)
	CourseFee	(SSPO)
	OnlineYN	(SSPF)
	RequiredForMajorTransition	(SSPF)

EMPLOYEE: It is any entity who works for CS Department.

Some of the example instances are: Staff, Tutor, Faculty.

Attributes:	**EmployeeId**	(SSPF)
	SSN	(SSPF)
	Name (FN, MN, LN)	(CSPO)
	Qualification	(SMPO)
	Role	(SSPF)
	Email	(SSPF)
	Salary	(SSPO)
	WorkExperience	(SSPO)

FACULTY: It is an entity, who teaches the courses and responsible for courses to be held. It will have all the attributes of Employee with additionally below ones.

Some of the example instances are: AI Prof, DBMS Prof .

Attributes:	Specialization	(SSPF)
	ResearchInterest	(SSPO)

INCIDENT: These are used by department to follow up the students to overcome the hurdles they faced.

Examples: Racial discrimination, Body shaming, Others etc.

Attributes:	**IncidentCategory**	(SSPF)
	Description	(SSPF)

JOB: It is an entity, student possess through which they earn wages and experience.

Some examples are: Software Engineer, Testing, DBMS Administrator.

Attributes:	**JobId**	(SSPF)
	JobType	(SSPF)
	Role	(SSPF)
	JobDescription	(SSPO)
	HourlyPay	(SSPO)
	OrganizationName	(SSPF)
	OrganizationInfo(Mail,Ph,Add)	(CSPO)

PROGRAM: It is a structure that a student pursues to complete the graduation.

Some examples are: Bachelors, MS, Ph.D..

Attributes:	**ProgramType**	(SSPF)
	Concentration	(SSPF)
	ProgramDescription	(SSPO)
	TotalCreditsNeeded	(SSPF)
	CoreCreditsNeeded	(SSPF)

RESEARCH_PROJECT: It is an entity that students and faculty work on, may be a part of students curriculum.

Some examples are: Building Drone, Face Recognitions.

Attributes:	**ProjectName**	(SSPF)
	FundsGranted	(SSPF)
	FundsUsed	(SSPO)
	CreditsForStudent	(SSPF)

STUDENT: A student is a person enrolled in educational institution who attend classes to complete graduation.

Some examples are: CS student, CE student, MIS student.

Attributes:	**StudentId**	(SSPF)
	StudentName(FN, MN, LN)	(CSPF)
	DateOfBirth	(SSPO)
	InternationalYN	(SSPF)
	Pre_majors_YN	(SSPO)
	Majors_YN	(SSPO)
	UnderRepresentedGroupName	(SSPO)
	Email	(SSPF)
	Phone	(SSPO)
	Address(Street, Apt, City, State,Zip)	(CSPO)
	ResumeLocation	(SSPF)

STAFF: Who works in a department but does not teach. It will have all the attributes of Employee with the role being type of staff.

Some examples are: Administrative staff, Technical.

SURVEY: It is a methodology of study taken from the participants to further rectify the mistakes or develop an organization.

Some Examples are: Pre-major survey, Major Survey, Exit Survey

Attributes:	**SurveyType**	(SSPF)
	Question	(SMPF)

TUTOR: Is a person who assists students in a specific course. It will have all the attributes of Employee with the role being tutor.

Some examples are: DBMS tutor, DIP tutor.

ON_CAMPUS: Jobs which are posted by organizations that are in campus and run by the University. Job type can be Student Job, Assistantships and all other except Tutor. It will have all the attributes of JOB with the JobType as on_campus. Role will change based on the role of the student

Some examples are: Graduate assistant, Teaching Assistant or Student Technical Specialist.

INTERNSHIP: Jobs which are posted by organizations other than University. It will have all the attributes of JOB with the JobType as internship and there role in the organization.

Some examples are: Software Engineer at RiskSense, Presbyterian IT support.

\

AWARD: Students can receive various awards such as scholarships or appreciations . Awards are differentiated using an attribute Type.

Attributes:	**AwardName**	(SSPF)
	AwardType	(SSPF)
	Appreciation	(SSPF)

2.2 Relationships

The relationships in this schema are described below:

taught_by: It is used establish the courses taught by each faculty.

Atributes:	Semester	(SSPF)
	Year	(SSPF)
	LectureHallName	(SSPO)
	LectureRoomNumber	(SSPO)
	StartTime	(SSPO)
	EndTime	(SSPO)

advisement: This relation-ship is helpful to know the which faculty is the academic advisor for which student, each semester.

Ex**:** academic advisement, faculty mentoring for underrepresented group student

Atributes:	Semester	(SSPF)
	Year	(SSPO)
	AdvisementType	(SSPF)

administrative_advisement: This relation-ship is helpful to know the which faculty is the administrative advisor for each student, each semester.

Atributes:	Semester	(SSPF)
	Year	(SSPO)

tutoring: With this relation-ship, we can know which tutor is assigned to which course and who all students use the tutoring services provided by the department.

pre_requisites: It is a self-referencing relation between courses, to show the pre-requisites for any course.

Attributes:	MinimumGradeForPreRequisite	(SSPF)

requirements: This relation-ship is used to know the courses the students need to take to complete a program of study.

enrolled_on: With this relation-ship, we can keep track the courses enrolled by the student till date and their status. It also useful to get the *academic transcrip*t at any time.

Atributes:	EnrolledSemester	(SSPF)
	EnrolledYear	(SSPF)
	StartDate	(SSPF)
	EndDate	(SSPO)
	Grade	(SSPF)
	Status	(SSPF)

opts_for: This relationship is useful to know the department program opted by the student.

Ex: Masters, Bachelors.

Atributes:	OptedSemester	(SSPF)
	OptedYear	(SSPF)
	StartDate	(SSPF)
	EndDate	(SSPO)
	Status	(SSPF)
Derived Attributes :		
	OverallGrade	(SSDF)

OverallGrade for each program = Sum of grades/ Number of Credits.

does: Students can work in various jobs. This relation-ship is helpful to keep track of all the jobs taken by students.

Atributes:	StartDate	(SSPF)
	EndDate	(SSPO)
	NumberofHoursPerWeek	(SSPF)

peer_mentoring: This relation-ship is helpful to know the which student is the mentor for other students from underrepresented groups, each semester.

Atributes:	Semester	(SSPF)
	Year	(SSPO)

gives: Department takes survey from students such as pre-major, transition and exit survey. This relation-ship is used to keep track of the surveys provided by each student.

Atributes:	SurveyDate	(SSPF)
	Rating	(SSPF)
	Comment	(SSPO)

participates: This relationship the holds the student participation in various activities conducted by the department.

Atributes:	ActivityDate	(SSPF)
	StartTime	(SSPF)
	Venue	(SSPF)
	Duration	(SSPF)

follow_up: This relationship keeps track of various incidents student face. Department will try to help to them to overcome the hurdles.

Atributes:	IncidentDate	(SSPF)
	NecessaryAction	(SSPF)

part_of: With this relationship, we can track which student is part of which research project.

Atributes:	StartDate	(SSPF)
	EndDate	(SSPO)
	NumberofHoursPerWeek	(SSPO)
	NumberofPapersPublished	(SSPO)

guides: With this relationship, we can track which faculty is guides which research project.

Atributes:	StartDate	(SSPF)
	EndDate	(SSPO)
	NumberofHoursPerWeek	(SSPO)
	NumberofPapersPublished	(SSPO)

works_as: It is used to keep track of student working as tutors.

Atributes:	EmploymentStartDate	(SSPF)
	EmploymentStartDate	(SSPO)
	NumberofHoursPerWeek	(SSPF)

helps: This relation is useful to keep log of which tutor helps which student in which course.

Atributes:	Semester	(SSPF)
	Year	(SSPO)
	FeedbacktoStudent	(SSPF)

achieves: This relation is useful to capture the awards received by each students.

Attributes :	Semester	(SSPF)
	Year	(SSPF)
	AppreciationReceived	(SSPF)

2.3Possible extensions and additional comments

- Student activities in other departments is not considered and can be taken care in further extensions.
- Department connections with Alumni students is not considered and can be taken care in further extensions.
- All the efforts by the department are considered, but not the further outcomes and actions.
- Steps taken by the department to handle the incidents with students have not been considered and can be included in the further extensions.

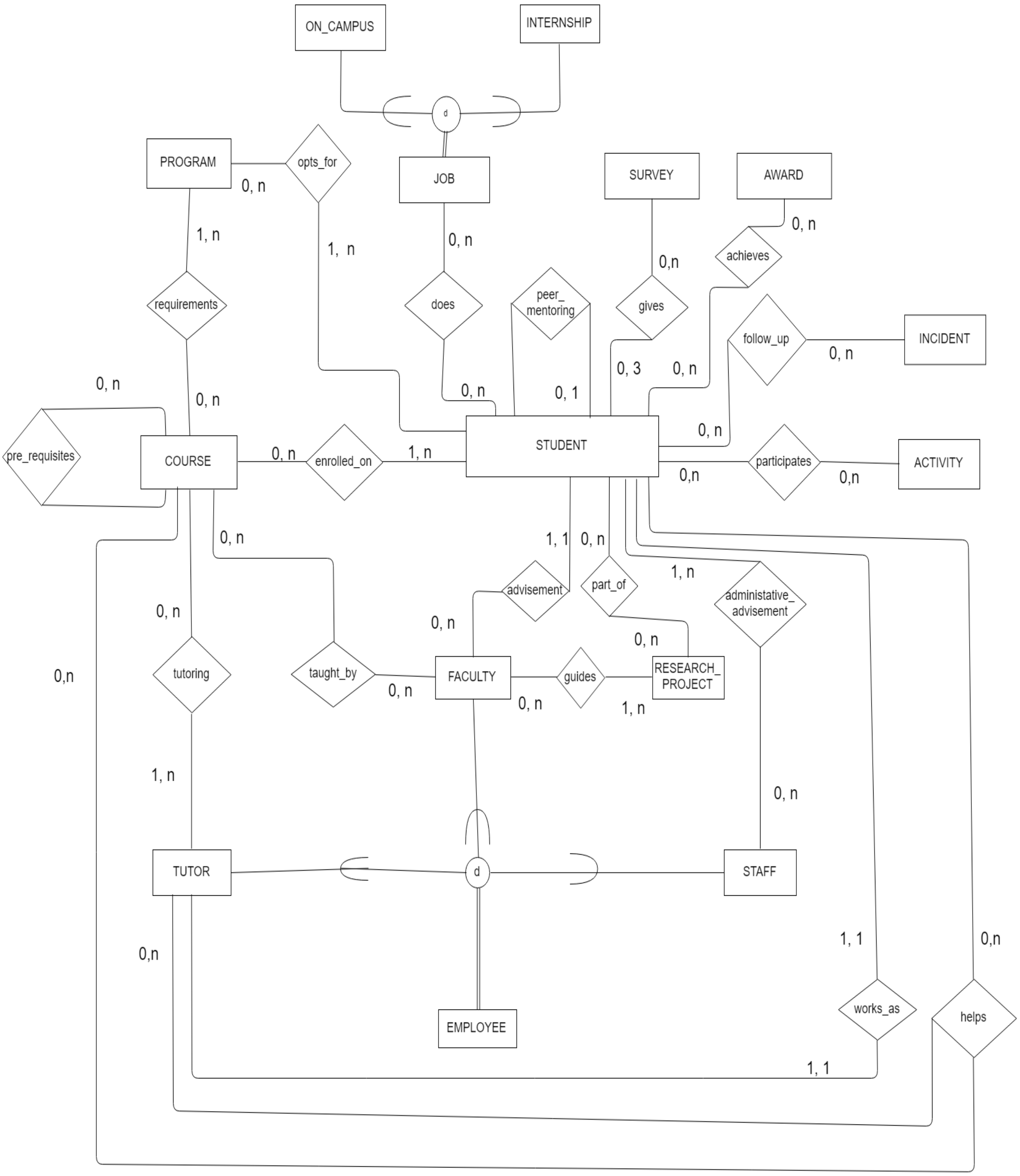

Figure 1 : CSRR Database

2.4 Explicit Integrity Constraints

Some examples of integrity constraints in our working example of a department database.

1. All numerical values must be positive.
2. A student can do internship only off campus.
3. A Tutor can be both student and can be hired from outside.
4. Research projects are not part of job.
5. A tutor is not a part of job.
6. A student from Under represented group can have either peer mentor or faculty mentor.
7. Student pursuing Pre-majors will have flag value 1 for pre majors, pursuing Majors have 1 in Majors category and having 1 in both indicates transition from pre-majors to majors.
8. An International student can work Off campus after 1 year.
9. Credits per semester required to complete program varies for international and national students.
10. Survey type can only be Pre-major, Major and Exit.
11. Activity can be of any of three types Social, Recruitment and Promotional.
12. For transition from pre-majors to majors or masters to Ph.D. few courses are compulsory.
13. Advisement relation has an attribute type which can only be either academic advisement or faculty mentoring for underrepresented group student.
14. Assuming minimum number of credits for an under graduate per year is 12, for graduate it.
15. Semester can have only three values namely Fall, Spring and summer.
16. Core courses should have a minimum grade of B, else the credit won’t be considered.
17. Course Start Date < Course End Date
18. Student can opt for only one program at a time in the department.
19. Course cannot have the same as a pre-requisite.
20. Program complete date< visa expiry for international students.
21. Tutor is only part of department employees and should not repeated in jobs.
22. Research projects are handled by department and not part of jobs.
23. All employees should have an SSN.

3. Example Queries

A list of the most important queries

1. Who all are faculty part of research projects?
2. How many credits do we need to complete master’s program?
3. What is percentage of students transition from pre-major to major this year?
4. List of all tutors in the department
5. List of faculty who joined in the last two years.
6. Activity name with highest percentage student participation.
7. List of students who work off-campus more than 20 hours ae week.
8. Get the academic transcript for given student ID.
9. What is most opted program of study this year?
10. List of all courses with no pre-requisites.

4.Mapping the CSRR Database to the Relational Model

The conceptual schema described for the CSRR Database is mapped into the Relational Schema presented in this section. Primary keys are underlined.

All the attributes underlined in the same Relation belong to the primary key.

STUDENT(StudentUniversityID, FirstName, MiddleName, LastName, DateofBirth, InternationalYN, PreMajorsYN, MajorsYN, UnderRepresentedGroupName, Email, Phone, Street, Apartment, City, State, Zipcode, ResumeLocation)

PROGRAM(ProgramType,Concentration,Description,TotalCreditsNeeded,TotalCoreCreditsNeeded**)**

COURSE(CourseCode , CourseName, CreditHours, Department, CourseFee, OnlineYN, RequiredForMajorTransition**)**

EMPLOYEE(EmployeeID, SSN, FirstName, MiddleName, LastName, Role, Email, WorkExperience, Salary**)**

EMPLOYEE_QUALIFICATION(EmployeeID, Degree, AcquiredDate, InstitutionName**)**

EmployeeID is foreign key, references EMPLOYEE

FACULTY(EmployeeID, Specialization, ResearchInterest**)**

EmployeeID is foreign key, references EMPLOYEE

ACTIVITY(ActivityName, ActivityType, Organizedby, OrganizerMail, OrganizerPhone**)**

JOB(JobID, JobType, Role, Description, HourlyPay, OrganizationName, Phone, Email, Street, Apartment, City, State, Zipcode**)**

SURVEY(SurveyType, Question**)**

INCIDENT(IncidentCategory, Description**)**

AWARD(AwardName, AwardType, Appreciation**)**

enrolled(StudentUniversityID, CourseCode, Semester, Year, StartDate, EndDate, CourseStatus, Grade**)**

StudentUniversityID is foreign key, references STUDENT

CourseCode is foreign key, references COURSE

pre_requisites(CourseCode, Pre-RequisiteCourseCode, MinimumGradeForPreRequisite**)**

CourseCode is foreign key, references COURSE

opts_for(StudentUniversityID, ProgramType, Concentration, OptedSemester, OptedYear, StartDate, EndDate, OverallGrade, Status**)**

StudentUniversityID is foreign key, references STUDENT

ProgramType, Concentration is foreign key, references PROGRAM

program_requirements(ProgramType, Concentration, CourseCode, MinimumGradeRequired**)**

CourseCode is foreign key, references COURSE

ProgramType, Concentration is foreign key, references PROGRAM

taught_by(CourseCode , EmployeeID, Semester, Year, LectureHallName, LectureRoomNumber, StartTime, EndTime**)**

EmployeeID is foreign key, references EMPLOYEE

CourseCode is foreign key, references COURSE

achieve(Semester, Year, AppreciationRecieved**)**

tutor_student(EmployeeID, StudentUniversityID**)**

EmployeeID is foreign key, references EMPLOYEE

StudentUniversityID is foreign key, references STUDENT

advisement(StudentUniversityID, EmployeeID, AdvisementType, Semester, Year**)**

tutoring(EmployeeID, CourseCode**)**

EmployeeID is foreign key, references EMPLOYEE

CourseCode is foreign key, references COURSE

does(StudentUniversityID, JobID, NumberofHoursPerWeek, StartDate, EndDate**)**

StudentUniversityID is foreign key, references STUDENT

JobID is foreign key, references JOB

gives(StudentUniversityID, SurveyType, QuestionName, SurveyDate, Rating, Comment**)**

StudentUniversityID is foreign key, references STUDENT

SurveyType, QuestionName is foreign key, references SURVEY

student_participation(StudentUniversityID, ActivityName, ActivityType, ActivityDate, StartTime, Venue, Duration**)**

StudentUniversityID is foreign key, references STUDENT

ActivityName, ActivityType is foreign key, references ACTIVITY

follow_up(StudentUniversityID, IncidentType, IncidentDate, IncidentTime, NecessaryAction**)**

StudentUniversityID is foreign key, references STUDENT

research_projects(ProjectName, FundsGranted, FundsUsed, CreditsForStudent**)**

part_of(StudentUniversityID, ProjectName, StartDate, EndDate, HoursPerWeek, NumberofPapersPublished**)**

StudentUniversityID is foreign key, references STUDENT

project_guide(StudentUniversityID, EmployeeID, StartDate, EndDate, HoursPerWeek, NumberofPapersPublished**)**

StudentUniversityID is foreign key, references STUDENT

EmployeeID is foreign key, references EMPLOYEE

helps(StudentUniversityID, EmployeeID, CourseCode, Semester, Year, FeedbacktoStudent**)**

EmployeeID is foreign key, references EMPLOYEE

StudentUniversityID is foreign key, references STUDENT

CourseCode is foreign key, references COURSE

student_peer_mentor(StudentUniversityID, MentorID, Semester, Year**)**

StudentUniversityID is foreign key, references STUDENT

4.1 Additional Integrity Constraints

1. *Concentration* in *PROGRAM* can be general or specific such as Artificial Intelligence, Databases etc.
2. *Role* in *EMPLOYEE* can be a staff or Faculty(Professor, assistant professor) or Tutor, but cannot be empty.
3. *JobType* can only be either on-campus or Internship.
4. *SurveyType* can only be either Pre-major, major, exit.
5. Each STUDENT can acquire only below *Grade* for the course taken.
   i. A, B, C, D ,F(fail), W(Withdrawn), Credit, Audit

6. *TotalCreditsNeeded*, *TotalCoreCreditsNeeded* >0 in *PROGRAM.*
7. *ActivityType* can only be Recruitment, Social or promotional.
8. *AdvisementType* can only be Administrative, academic, or Mentoring
9. Any *StartDate<EndDate*
10. *Duration*>0
11. Funds granted and funds used for research projects must be postivie,
12. *CourseFee* >=0
13. *JOB's HourlyPay>0* and *NumberofHoursPerWeek*>0.
14. Every *SURVEY* must have *Rating* between 0 and 5.
15. Each *STUDENT* must have atleast one tuple in the relation *ENROLLED_ON* and maximum one tuple for giver *Semester* and *Year*.
16. Each *STUDENT* must have atleast one tuple in the relation *ADVISEMENT* and maximum one tuple for giver *Semester* and *Year*.
17. *AwardType* can be either Academic, Scholarship, Extracurricular etc.
18. Each *EmployeeID* in *TUTORING* must have the *Role* as Tutor in *EMPLOYEE.*
19. Every *EMPLOYEE* of *Role* Tutor must have atleast one tuple in the relation *TUTORING.*
20. Every *EMPLOYEE* of *Role* Tutor must have atleast and at maximum one tuple in the tuple *WORKS_AS.*
21. Each *RESEARCH_PROJECT* must have atleast one tuple in the relation *GUIDES.*
22. Every STUDENT must have atleast one tuple in the relation *OPTS_FOR.*
23. Attribute *Semester* can only be either Fall, Spring or Summer.
24. Attribute *NumberofPapersPublished* in the relations *PART_OF* or *GUIDES* must be greater than or equal to 0.
25. A *STUDENT* can have only either faculty mentor or student mentor for the given *Semester* and *Year*.
26. For any *STUDENT* at any time, list of all tuples in the relation *ENROLLED_ON* gives the academic transcript.
27. *PreMajorYN* and *MajorsYN* can have only values either true or false which decides they have opted for Undergraduate and graduate .