**Students** (

Aid: LONG [PK, FK to Accounts.id],

DIsplayedName: VARCHAR(50),

FirstName: VARCHAR(50),

LastName: VARCHAR(50),

Major: VARCHAR(50),

Year: INT, -> bachelor 4 / master / phd

PreferredWorkTime: INT,

)

Course\_taken(

FK Aid

FK CRN

Taken/taking INT (0/1 0-taking, 1-taken)

)

**Accounts** (

id: LONG [PK],

userName: VARCHAR(64)

Credential: LONG

)

**Projects** (

id: LONG [PK],

AccountID: LONG [FK to Accounts.id],

)

**Courses** (

CRN: LONG [PK]

CourseNumber: INT

Department: VARCHAR(50)

// GroupchatContact: VARCHAR(50)

)

Course\_Group(

Cid AUTO\_INCREMENT PK,

FK CRN

INT Type(wechat, discord, campuswire)

Entry VarChar

)

**Requests**(

Source: LONG [Student.AccountId],

Target: LONG [Student.AccountId],

Message: VARCHAR(1024)

)

**Connections** (

(undirected)

self: LONG [Student.AccountId],

other: LONG [Student.AccountId],

)

**Project\_Static (**

pid: LONG [FK to Projects.id],

AssignmentType: INT ( Assignment, Course Project, Independent)

CRN: Foreign key (Courses.CRN)

StartDate: DATE

EndDate: DATE

GroupCapacity: INT,

AssignmentDescription: VARCHAR(255),

CaptainId: Foreign key(Students.aid),

(Other assessment information…)

**)**

**Favourites (**

Aid FK,

Pid FK

**)**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

CREATE TABLE Student (

NetId: VARCHAR(10),

AccountID: LONG ,

NickName: VARCHAR (50),

FirstName: VARCHAR(50),

LastName: VARCHAR(50),

Department: VARCHAR(50),

PreferredWorkTime: VARCHAR(50),

SchoolYear: VARCHAR(50),

ContactInformation: VARCHAR(255)

PRIMARY KEY AccountID,

FOREIGN KEY REFERENCES Students (Aid),

ON DELETE CASCADE,

ON UPDATE CASCADE

)

CREATE TABLE Courses (

CRN INT,

CourseName VARCHAR(10),

Department VARCHAR(50),

GroupchatContact VARCHAR(50),

PRIMARY KEY (CRN)

);

CREATE TABLE Team\_Static (  
 TId: INT,  
 AssessmentType: VARCHAR(50),  
 AssesmentCRN: INT,  
 GroupCapacity: INT,  
 AssessmentDescription: VARCHAR(255),  
 CaptainNetId: VARCHAR(10),  
 Active: BOOLEAN,  
 PRIMARY KEY (TId),  
 FOREIGN KEY (AssesmentCRN) REFERENCES Courses (CRN),  
 ON DELETE CASCADE,  
 ON UPDATE CASCADE,  
 FOREIGN KEY (CaptainNetId) REFERENCES Student(NetId),  
 ON DELETE CASCADE,  
 ON UPDATE CASCADE

)