**CS 220**

**Database Systems**

**Final Project Report**

**MadLabs**

Submitted to: Ms. Hirra Anwar

Date of submission: 16-01-2017

Class: BESE 6A

Submitted by:

|  |  |  |
| --- | --- | --- |
| Name | Registration number | Email |
| Maryam Sulemani | 00000 | msulemani.bese15seecs@seecs.edu.pk |
| Wajeeha Yasin | 00000120012 | wyasin.bese15seecs@seecs.edu.pk |
| Duaa Zahra Khan | 00000137700 | dkhan.bese15seecs@seecs.edu.pk |

Project Proposal

* **Domain and scope:**

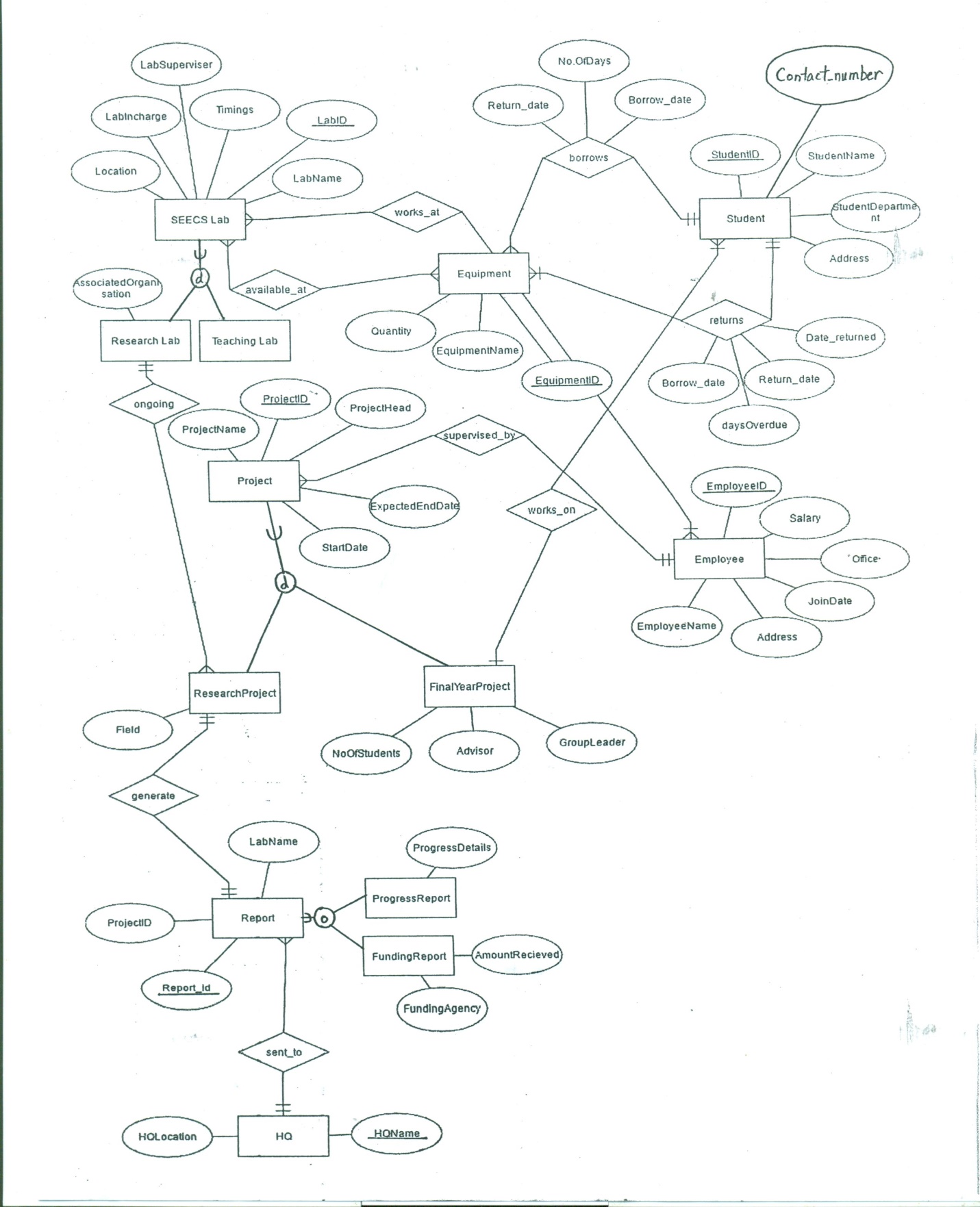
The domain of our project is inventory management in the labs of NUST H-12.

We wish to create MadLabs - a web application that allows the university to manage the record, status and information of the equipment and resources in the labs. We also aim to implement a system that allows the students to use and borrow resources from all the labs in the campus irrespective of the department they belong to.

* **Functional requirements:**
* stores information about the status and availability of existing resources
* maintains the record of orders
* secure portal for students to check the available equipment and the duration of availability
* stores a defined criteria for which issuance is allowed
* maintains a record of authorized people for expensive equipment
* records the requests made by students for ordering new equipment
* **Non-functional requirements:**
* give authorization to modify data only to the concerned authority
* should be able to store large amount of data
* should be fast in retrieving data
* should take back up of the data to avoid any losses in case the system crashes
* **Motivation behind this project:**

Our main motivation behind this idea was to be able to make the resources available in the campus more accessible to the students. This will help them increase the scope and enhance the quality of their projects. It will also help the authorities track the equipment hence removing any concerns over the safety of the equipment.

Entity Relation Ship Diagram

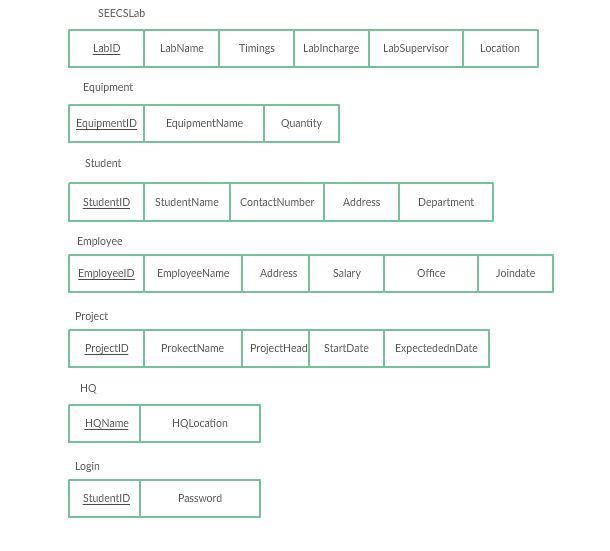


Business Rules

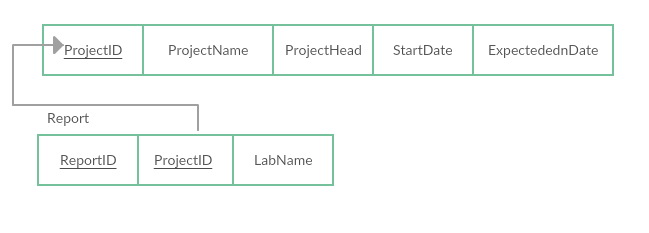
* EQUIPMENT is available at SEECS LAB. EQUIPMENT has a name, an ID and the quantity available. A SEECS LAB has a name, lab ID, a supervisor, an instructor, location and the timings for which it is opened.
* A SEECS LAB can be a TEACHING LAB or a RESEARCH LAB. A RESEARCH LAB may have an organization associated with it.
* STUDENT can borrow EQUIPMENT. A STUDENT has student ID, name, department, class, contact number and address.
* EQUIPMENT can only be borrowed by one STUDENT at a time.
* STUDENT must return the borrowed EQUIPMENT on time.
* EMPLOYEE works in a lab. EMPLOYEE has employee ID, name, salary, join date and address.
* EMPLOYEE supervises PROJECT. PROJECT has a project ID, name, project head, start date and end date. A PROJECT may be a RESEARCH PROJECT or a FINAL YEAR PROJECT. RESEARCH PROJECT has a field. FINAL YEAR PROJECT has an advisor, group leader and number of students.
* STUDENT works on FINAL YEAR PROJECT.
* RESEARCH LAB has REASEARCH PROJECT.
* A biannual REPORT of each RESEARCH PROJECT must be generated. REPORT has a report ID. REPORT may be a PROGRESS REPORT or a FUNDING REPORT. PROGRESS REPORT has progress details. FUNDING report has a funding agency and amount.
* The REPORT must be sent to the HEADQUARTER. HEADQUARTER has a name and location.

Relational Schema

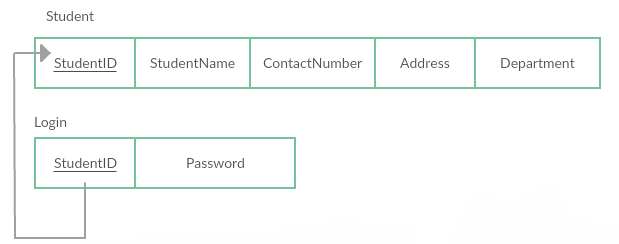
**Step1: Mapping of regular entity types**

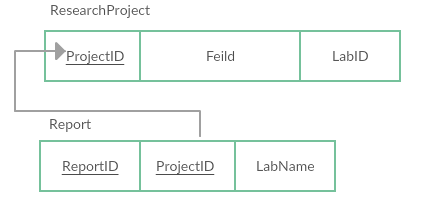


**Step2: Mapping of weak entity type**

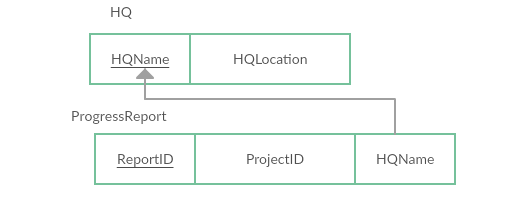


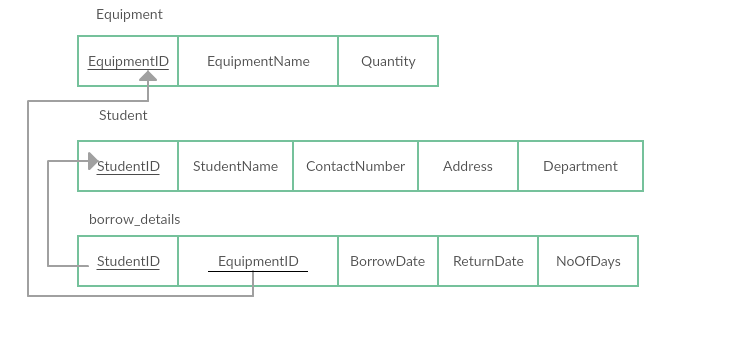
**Step3: Mapping of binary, 1-1 relationship types**

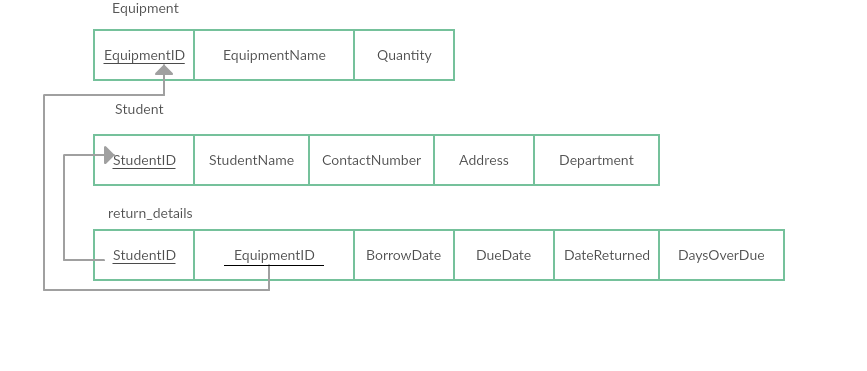


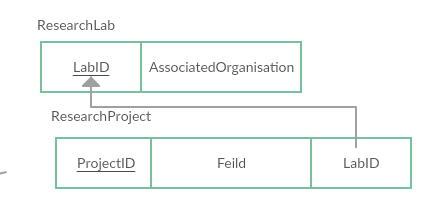
**~~~~**

**Step4: Mapping of binary, 1-n relationship types**

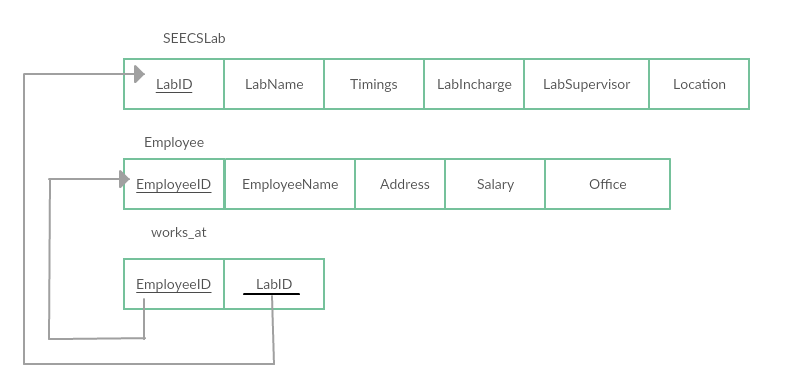


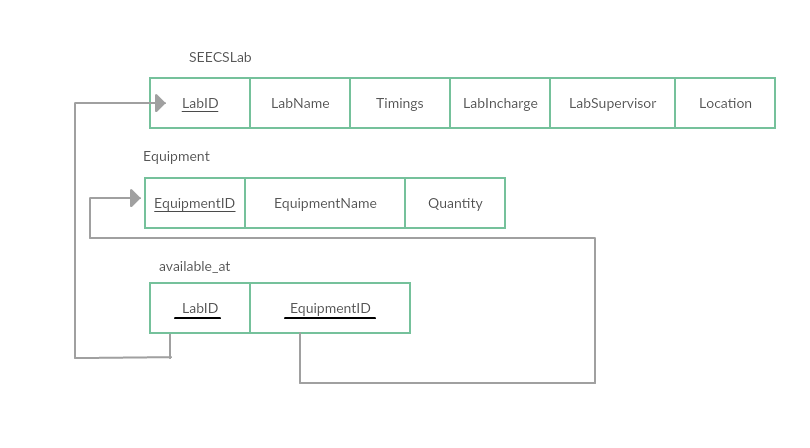


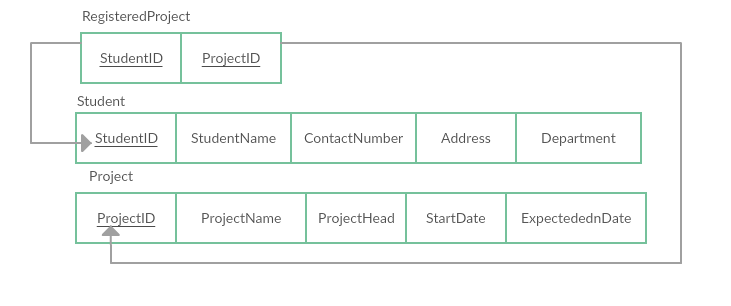




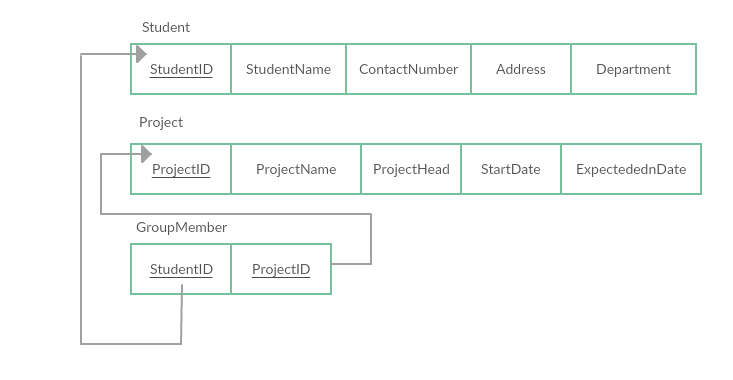
**Step5: Mapping of binary, M-N relationship types**







**Step 6: Mapping of Multivalued Attributes:**

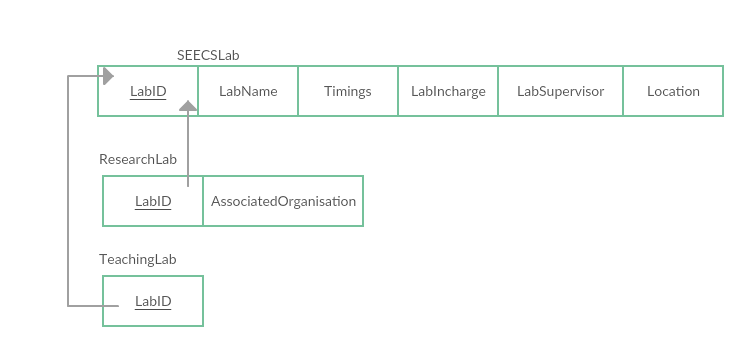


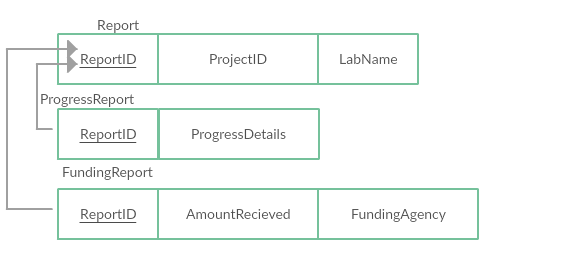
**Step7: Mapping of *N*-ary Relationship Types**

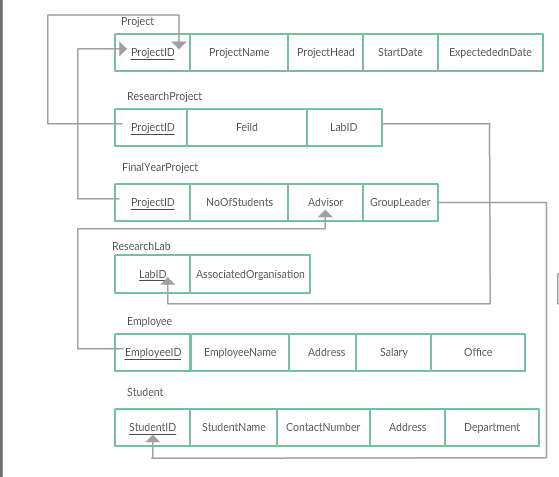
None

**Step8: Options for Mapping Specialization or Generalization**

**Option 8A: Multiple relations—superclass and subclasses**

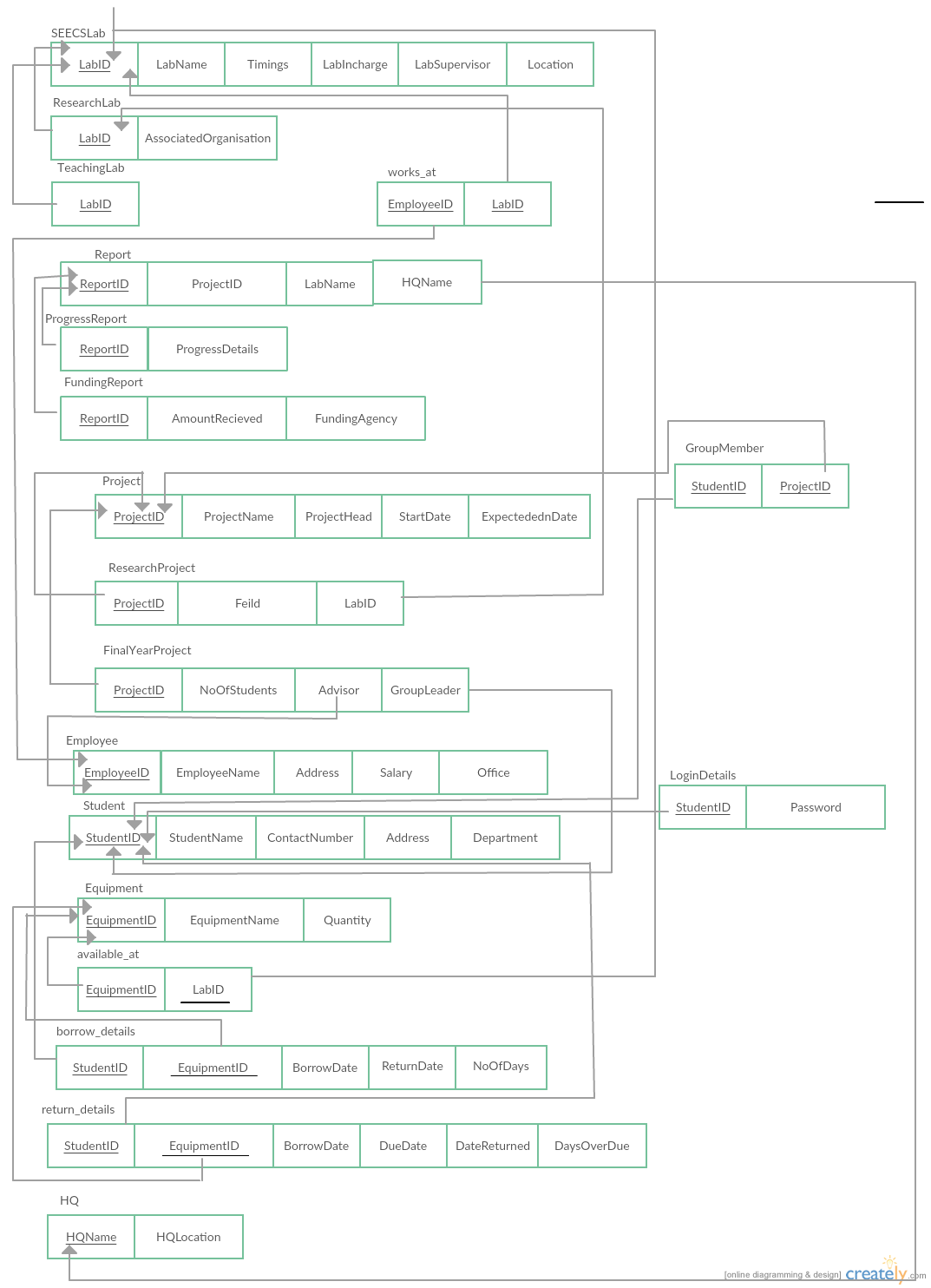
**~~~~**

**~~~~**

**~~~~**

**Step9: Mapping of Union Types**

None

**Final Schema:**

Normalized Schema

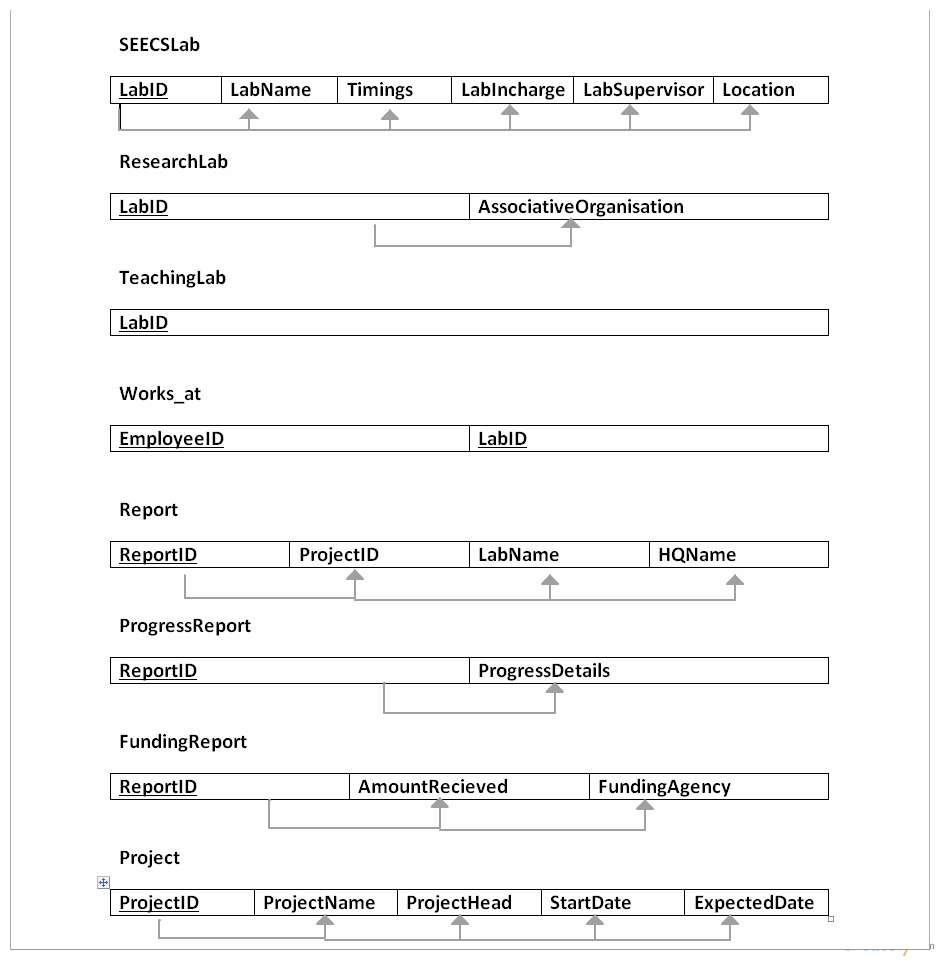
**1st Normal Form**

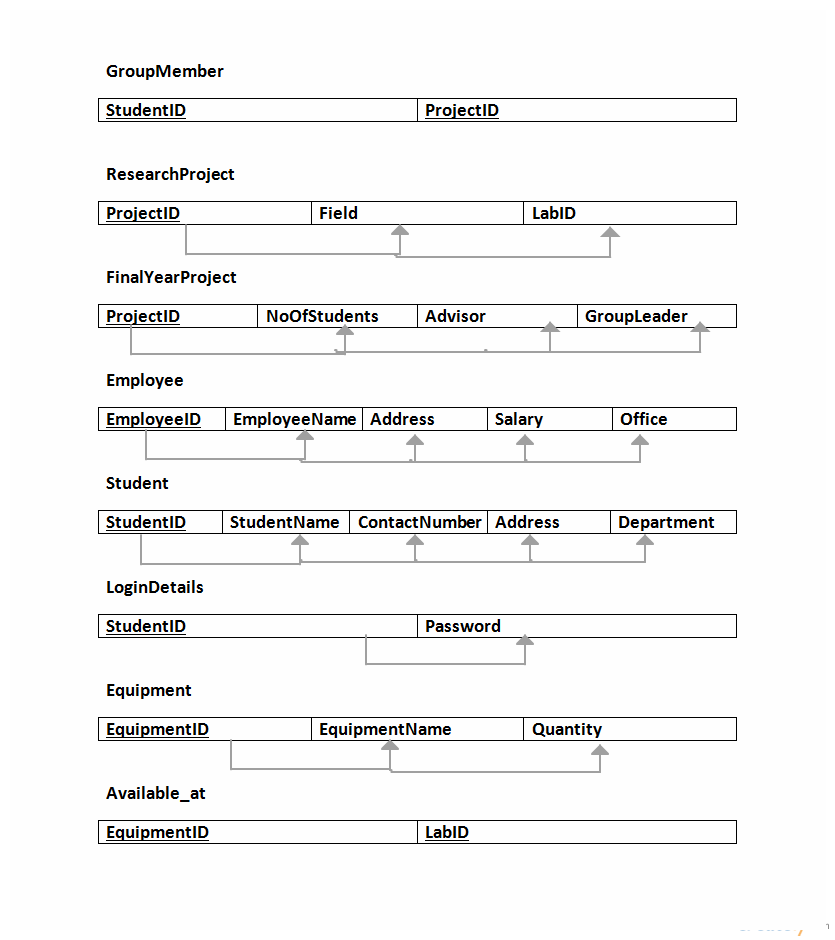
Our schema is already in first normal form as there exists no multivalued attribute in any of our relation.

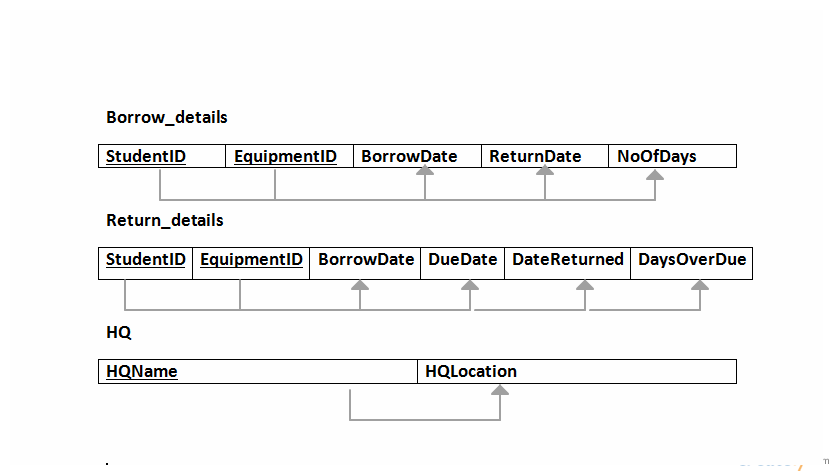
**2nd Normal Form**

**SHOWING ALL THE FUNCTIONAL DEPENDENCIES**

Our schema is already in 2NF because all the functional dependencies that exist are full functional dependencies and majority of relations contain only a single attribute in their primary key.

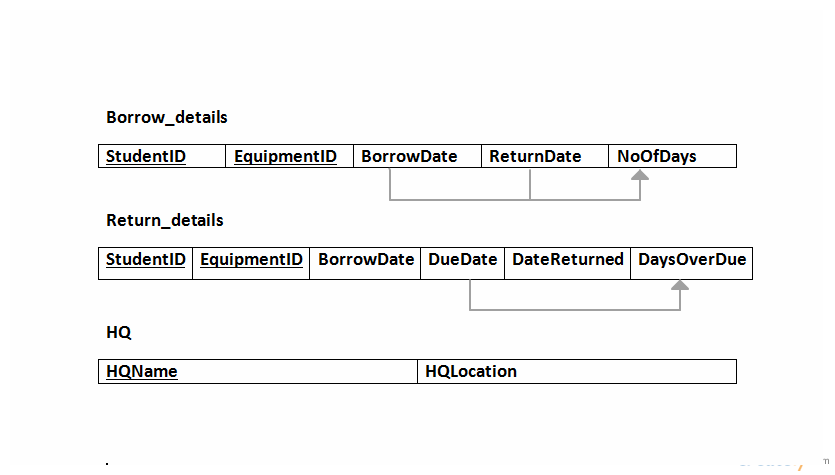
****

****

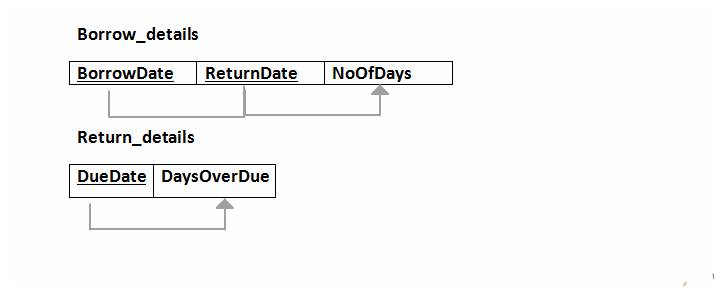
****

**3RD Normal Form**

For our schema to be in 3NF, there should be no Transitive Functional Dependencies but following transitive functional dependencies exists in the following part of our schema.



**Converting into 3NF**

****

**Due\_Days**

**Borrow\_Duration**

Data Definition Language

CREATE TABLE Employee

(EmployeeID INT,

  Salary INT ,

  Office VARCHAR(50) ,

  EmployeeName VARCHAR(50) ,

  JoinDate date ,

  PRIMARY KEY (EmployeeID)

);

CREATE TABLE HQ

(

  HQLoaction VARCHAR(50) NOT NULL,

  HQName VARCHAR(50),

  PRIMARY KEY (HQName)

);

CREATE TABLE Equipment

(

  EquipmentID INT ,

  EquiomentName VARCHAR(50) ,

  Quantity INT ,

  PRIMARY KEY (EquipmentID)

);

CREATE TABLE SEECSLab

(

  LabID INT ,

  Location VARCHAR(50),

  Timings time ,

  LabName INT ,

  LabIncharge INT,

  LabSupervisor INT ,

  PRIMARY KEY (LabID),

  FOREIGN KEY (LabIncharge) REFERENCES Employee(EmployeeID),

  FOREIGN KEY (LabSupervisor) REFERENCES Employee(EmployeeID)

);

CREATE TABLE ResearchLab

(

  AssociatedOrganisation INT ,

  LabID INT ,

  PRIMARY KEY (LabID),

  FOREIGN KEY (LabID) REFERENCES SEECSLab(LabID)

);

CREATE TABLE TeachingLab

(

  LabID INT,

  PRIMARY KEY (LabID),

  FOREIGN KEY (LabID) REFERENCES SEECSLab(LabID)

);

CREATE TABLE Project

(

  ProjectName INT,

  ProjectID INT ,

  ExpectedEnddate date ,

  StartDate date ,

  ProjectHead INT ,

  PRIMARY KEY (ProjectID),

  FOREIGN KEY (ProjectHead) REFERENCES Employee(EmployeeID)

);

CREATE TABLE ResearchProject

(

  Field VARCHAR(50),

  ProjectID INT,

  LabID INT ,

  PRIMARY KEY (ProjectID),

  FOREIGN KEY (ProjectID) REFERENCES Project(ProjectID),

  FOREIGN KEY (LabID) REFERENCES ResearchLab(LabID)

);

CREATE TABLE Student

(

  StudentID INT ,

  StudentName VARCHAR(50),

  ContactNumber INT,

  StudentDepartment VARCHAR(50) ,

  Address VARCHAR(50) ,

   PRIMARY KEY (StudentID)

);

CREATE TABLE Report

(

  ReportID INT ,

  HQName VARCHAR(50) ,

  ProjectID INT ,

  PRIMARY KEY (ReportID),

  FOREIGN KEY (HQName) REFERENCES HQ(HQName),

  FOREIGN KEY (ProjectID) REFERENCES ResearchProject(ProjectID)

);

CREATE TABLE ProgressReport

(

  ProgressDetails VARCHAR(200) ,

  ReportID INT ,

  PRIMARY KEY (ReportID),

  FOREIGN KEY (ReportID) REFERENCES Report(ReportID)

);

CREATE TABLE FundingReport

(

  AmountRecieved INT ,

  FundingAgency VARCHAR(50) ,

  ReportID INT ,

  PRIMARY KEY (ReportID),

  FOREIGN KEY (ReportID) REFERENCES Report(ReportID)

);

CREATE TABLE works\_at

(

  EmployeeID INT,

  LabID INT,

  PRIMARY KEY (EmployeeID, LabID),

  FOREIGN KEY (EmployeeID) REFERENCES Employee(EmployeeID),

  FOREIGN KEY (LabID) REFERENCES SEECSLab(LabID)

);

CREATE TABLE available\_at

(

  LabID INT,

  EquipmentID INT,

  PRIMARY KEY (LabID, EquipmentID),

  FOREIGN KEY (LabID) REFERENCES SEECSLab(LabID),

  FOREIGN KEY (EquipmentID) REFERENCES Equipment(EquipmentID)

);

CREATE TABLE borrow\_details

(

  Borrow\_date date ,

  Return\_date date ,

  No\_of\_days INT ,

  StudentID INT ,

  EquipmentID INT ,

  PRIMARY KEY (StudentID, EquipmentID),

  FOREIGN KEY (StudentID) REFERENCES Student(StudentID),

  FOREIGN KEY (EquipmentID) REFERENCES Equipment(EquipmentID)

);

CREATE TABLE return\_details

(

  Borrow\_date date ,

  Due\_date date ,

  Date\_returned date ,

  Days\_overdue INT ,

  EquipmentID INT,

  StudentID INT,

  PRIMARY KEY (EquipmentID, StudentID),

  FOREIGN KEY (EquipmentID) REFERENCES Equipment(EquipmentID),

  FOREIGN KEY (StudentID) REFERENCES Student(StudentID)

);

CREATE TABLE RegisteredProject

(

  StudentID INT ,

  ProjectID INT ,

  PRIMARY KEY (StudentID, ProjectID),

  FOREIGN KEY (StudentID) REFERENCES Student(StudentID),

  FOREIGN KEY (ProjectID) REFERENCES Project(ProjectID)

);

CREATE TABLE LoginDetails

(

  Password INT ,

  StudentID INT ,

  PRIMARY KEY (StudentID),

  FOREIGN KEY (StudentID) REFERENCES Student(StudentID)

);

CREATE TABLE FinalYearProject

(

  NoOfStudents INT ,

  ProjectID INT ,

  GroupLeader INT ,

  Advisor INT ,

  PRIMARY KEY (ProjectID),

  FOREIGN KEY (ProjectID) REFERENCES Project(ProjectID),

  FOREIGN KEY (GroupLeader) REFERENCES Student(StudentID),

  FOREIGN KEY (Advisor) REFERENCES Employee(EmployeeID)

);

CREATE TABLE GroupMember

(

  StudentID INT ,

  ProjectID INT ,

  PRIMARY KEY (StudentID, ProjectID),

  FOREIGN KEY (StudentID) REFERENCES Student(StudentID),

  FOREIGN KEY (ProjectID) REFERENCES FinalYearProject(ProjectID)

);

Data Insertion Queries

insert into progressreport values

("Half yearly report 2015",111);

insert into progressreport values

("Annual report 2015",112);

insert into progressreport values

("Half yearly report 2016",113);

insert into progressreport values

("Half yearly report 2015",114);

insert into hq values

("Scholar's Avenue","NUST Headquarters");

insert into works\_at values

(100,101);

insert into works\_at values

(200,102);

insert into works\_at values

(300,103);

insert into works\_at values

(400,104);

insert into works\_at values

(500,100);

insert into works\_at values

(600,101);

insert into works\_at values

(700,102);

insert into works\_at values

(800,103);

insert into works\_at values

(900,104);

insert into works\_at values

(1000,100);

insert into groupmember values

(4,1000);

insert into groupmember values

(3,1000);

insert into groupmember values

(1,1002);

insert into groupmember values

(2,1002);

insert into groupmember values

(5,1003);

insert into groupmember values

(6,1003);

insert into researchproject values

("Healthcare",1001,104);

insert into researchproject values

("Cloud Computing",1004,103);

(2,1000,4,100);

insert into finalyearproject values

(2,1002,1,500);

insert into finalyearproject values

(2,1003,5,200);

insert into available\_at values

(100,10);

insert into available\_at values

(100,11);

insert into available\_at values

(100,12);

insert into available\_at values

(101,13);

insert into available\_at values

(101,14);

insert into available\_at values

(101,15);

insert into available\_at values

(102,16);

insert into available\_at values

(102,17);

insert into available\_at values

(102,18);

insert into available\_at values

(103,19);

insert into available\_at values

(103,20);

insert into available\_at values

(103,21);

insert into available\_at values

(104,22);

insert into available\_at values

(104,23);

insert into available\_at values

(104,24);

insert into equipment values

(10,"Digital Logic Trainer",10);

insert into equipment values

(11,"IC Tester",5);

insert into equipment values

(12,"Digital Multimeter",12);

insert into equipment values

(13,"Oscilliscope",10);

insert into equipment values

(14,"TTL IC",40);

insert into equipment values

(15,"Lego Minstorm Kit",15);

insert into equipment values

(16,"Snap Circuit Kit",15);

insert into equipment values

(17,"Trainer Board",15);

insert into equipment values

(18,"Function Generator",20);

insert into equipment values

(19,"Thermocouple",5);

insert into equipment values

(20,"Pressure Sensor",3);

insert into equipment values

(21,"PLC Training Kit",4);

insert into equipment values

(22,"Ultra Sonic Level Sensor",2);

insert into equipment values

(23,"Pendulum Trainer",10);

insert into equipment values

(24,"Dual Display Digital Multimeter",5);

insert into teachinglab values(100);

insert into teachinglab values(101);insert into teachinglab values(102);

insert into researchlab values

("US-AID",103);

insert into researchlab values

("TUKL",104);

insert into logindetails values

(1111,1);

insert into logindetails values

(1222,2);

insert into logindetails values

(1333,3);

insert into logindetails values

(1444,4);

insert into logindetails values

(1555,5);

insert into logindetails values

(1666,6);

insert into logindetails values

(1777,7);

insert into student values

(1,"Maryam Sulemani",03331111,"DOC","F-7");

insert into student values

(2,"Wajeeha Yasin",03330000,"DOC","F-11");

insert into student values

(3,"Duaa Zahra Khan",03011111,"DOC","F-10");

insert into student values

(4,"Christiano Ronaldo",03001011,"DOC","E-8");

insert into student values

(5,"Sergio Ramos",03001111,"EE","E-8");

insert into student values

(6,"Luka Modric",03451111,"EE","H-12");

insert into student values

(7,"James Rodriguez",03021111,"DOC","H-13");

insert into employee values

(600, 50000, "A-100", "Farrukh Abbas",'2013-06-24');

insert into employee values

(700, 50000, "A-220", "Saqib Nazir",'2010-05-23');

insert into employee values

(800, 60000, "A-210", "G. Hussain Asim",'2011-06-20');

insert into employee values

(900, 75000, "A-310", "Sohaib Jamal",'2014-06-19');

insert into employee values

(1000, 70000, "A-300", "Aqsa Ahmad",'2009-05-30');

(100, 100000, "A-101", "Masroof Asif",'1986-10-05');

insert into employee values

(200,100000,  "A-104","Dr. Ammar Hassan", '2010-08-10');

insert into employee values

(300, 500000, "A-110","Dr. Hammad Cheema", '2008-10-10');

insert into employee values

(400, 400000, "A-102","Dr. Hassan Aqeel", '2012-05-15');

insert into employee values

(500, 200000, "A-105", "Prof. Arshad Nazir",'2013-06-23');