

**DBMS ASSIGNMENT 1**

**TITLE:EXAMPOINT**

**Name: K.Rushitha**

**Sec: IT B**

**Roll No:1602-19-737-090**

**VIRTUAL ASSESSMENT MANAGEMENT SYSTEM**

**(EXAM POINT)**

ABSTRACT:

***Virtual Assessment*** System is a web application that establishes a network between the server and the students. Here admin can be an institute (or server itself) enter on the site the questions they want in the exam. These questions are displayed as a test to the eligible students. The answers entered by the students are then evaluated and their score is calculated and saved. This score then can be accessed by the institutes or known to students if they want to display them or determine the passes students or to evaluate their performance and also one such special feature is that it records the attendance of the students who attended the exam.

***Virtual Assessment*** provides the platform where its participates in, and it involves the users also in any tests conducted. Questions are posted by the site, and also users of the site. The site requires an institute to register before posting the questions.

The site has an administrator who keeps an eye on the overall functioning of the system. The site gets revenue by charging the institutes each time they want to conduct the exam. The system entitled “EXAM POINT” is application software, which aims at providing services to the institutes and providing them with an option of selecting the eligible students by themselves. It is developed by using J2EE technology and related database.

**REQUIREMENT ANALYSIS :**

This project is a Java application that is developed in Eclipse having Oracle server as back end.

1. Database Design using Oracle

3. Coding (Oracle databse)

***Hardware Interface:***

**Client Side:**

Eclipse

Processor: Pentium IV 2.0 and above.

RAM: 256 MB

**Server Side:**

Processor: Pentium IV 2.0 and above.

RAM: 1 GB

Disk space: 4GB

**Tables:**

* Admin
* User
* Exam
* CoursesInfo
* QuestionInfo
* Attendance
* Answers
* Performance

List of attributes with their table heads:

**Admin:**

* Admin Password : Apassword - VARCHAR2(20)
* Admin’s login id : Aid - VARCHAR2(10)
* Admin User Name : Aname - VARCHAR(20)

**User\_Details:**

* Student Name : Sname - VARCHAR2(20)
* Student Registeration number : Sregno - VARCHAR2(20)
* Student Login ID : Sid - VARCHAR(5)
* Student’s Login Password : Spassword - VARCHAR(10)
* Last Name of the User : Lname - VARCHAR(10)
* First Name Of The Usr : Fname - VARCHAR(10)

**Course\_Info:**

* Course ID : Cid - VARCHAR2(10)
* Course Name : Cname - VARCHAR2(10)

**Questions:**

* Question ID : Qid - VARCHAR2(10)
* Question Name : Qname - VARCHAR2(200)
* Answer of the question ID : Ans - VARCHAR2(100)
* Option 1 : Opt1 - VARCHAR2(5)
* Option 2 : Opt2 - VARCHAR2(5)
* Option 3 : Opt3 - VARCHAR2(5)
* Option 4 : Opt4 - VARCHAR2(5)

**Answers**:

* Course ID : cid - VARCHAR (10)
* Question Name : qname - VARCHAR2(100)
* Correct Option : crct\_opt - VARCHAR2(5)

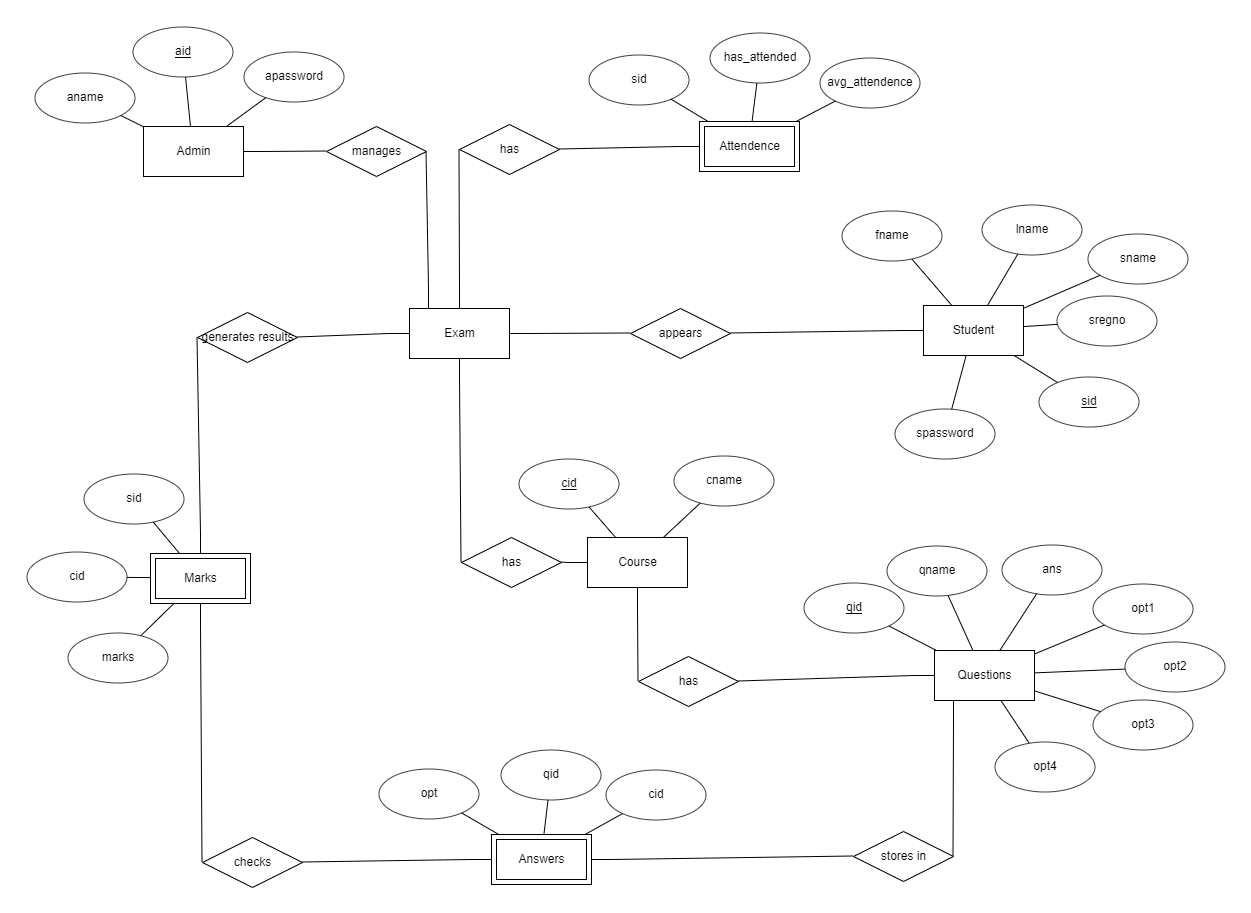
**Performance**:

* Total Mark : marks - NUMBER(5)
* Course ID : cid - VARCHAR2(10)
* Student ID : sid - VARCHAR2(5)

**Attendance**:

* Student ID : sid - VARCHAR2 (20)
* Has Attended : has\_attended - VARCHAR2(5)
* Has Avg Attended : avg\_attendance - FLOAT (5,2)

**E –R Diagram**:



**MAPPING CARDINALITY AND PARTICIATION**

**Constraints**

Here,the admin is connected to exam server.so there can be many admins who may be connected t0 server. So it is many to one relation,.

Similarly number of students attended for the exam will be recorded . So it is many to one relation.

No of courses in the exam will be taken and it is connected to the exam server .So it one to many relation.

There will be many number of courses involved and similarly many number of questions for each course . So it is many to many relation.

Similarly each question in the data base has only one correct anser so it is one to one relation.

Results will be evolved based on their courses for each student . So it is many to many relation.

**DDL COMMANDS:**

* create table admin(aid number(10),aname varchar2(20),apassword varchar2(20));
* create table user\_details(sid number(5),sname varchar2(20),spassword varchar2(20),sregno number(6),fname varchar2(10),lname varchar2(10));
* create table course\_info(cid number(10),cname varchar2(10));
* create table questions(qid number(10),qname varchar2(100),answer varchar2(20),opt1 varchar2(5),opt2 varchar2(5),opt3 varchar2(5),opt4 varchar2(5));
* create table answers(cid number(10),qname varchar2(100),crct\_opt varchar2(5));
* create table performance(marks number(5),cid number(10),sid number(5));
* create table attendance(sid number(5),has\_attended varchar2(5),avg\_att float(5));

**DML COMMANDS:**

* insert into admin values(&aid,’&aname’,’&apassword’);
* insert into user\_details values('&sid','&sregno','&sname','vce@123','&fname','&lname');
* insert into course\_info values('&cid','&cname');
* insert into questions values('&qid','&qname','&answer','&opt1','&opt2','&opt3','&opt4','&cid');
* insert into answers values('&cid','&crct\_opt','&qid');
* insert into attendance values('&sid','&has\_attended','&avg\_att');
* insert into performance values('&marks','&cid','&sid');

**Integrity Constraints:**

* alter table admin add PRIMARy key(aid);
* alter table user\_details add primary key(sid);
* alter table course\_info add primary key(cid);
* alter table questions add primary key(qid);
* ALTER TABLE questions ADD CONSTRAINT un\_ques\_answer UNIQUE(qname,answer);
* ALTER TABLE course\_info ADD CONSTRAINT un\_cname UNIQUE(cname);
* ALTER TABLE user\_details ADD CONSTRAINT un\_regno UNIQUE(sregno);
* ALTER TABLE admin MODIFY (aname varchar2(20) CONSTRAINT nn\_an NOT NULL);
* ALTER TABLE user\_details MODIFY (sname varchar2(20) CONSTRAINT nn\_sn NOT NULL,sregno varchar2(20) CONSTRAINT nn\_sr NOT NULL);
* ALTER TABLE questions MODIFY (qname varchar2(200) CONSTRAINT nn\_qn NOT NULL,answer varchar2(20) CONSTRAINT nn\_ar NOT NULL);
* alter table answers add foreign key(cid) references course\_info;

QUERY OUTPUTS:

