ST. XAVIER'S COLLEGE

**(Affiliated to Tribhuvan University)**

Maitighar, Kathmandu



**DATABASE MANAGEMENT SYSTEM**

**LAB ASSIGNMENT # 4**

**SUBMITTED BY:**

Dwarika Shiwakoti  
013BSCCSIT020

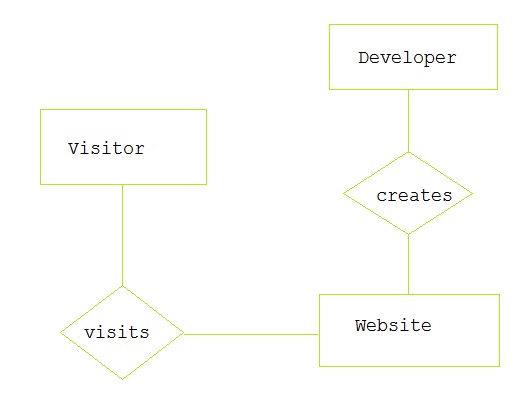
**SUBMITTED TO:**

|  |  |
| --- | --- |
| **Er. Sanjay Kumar Yadav**  Supervisor,  Department of Computer Science  St. Xavier’s College |  |

**Date of Submission:** 10th August 2015

**ER diagram with case study:**

ER-Diagram is a visual representation of data that describes how data is related to each other.



**Design**

**Database design** is the process of producing a detailed [data model](https://en.wikipedia.org/wiki/Data_model) of a [database](https://en.wikipedia.org/wiki/Database). This [logical data model](https://en.wikipedia.org/wiki/Logical_data_model) contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in a [data definition language](https://en.wikipedia.org/wiki/Data_definition_language), which can then be used to create a database. A fully attributed data model contains detailed attributes for each entity.

**Functional Design:**

**Database Design :**

**ER to relational mapping:**

A relation is a table that holds the data we are interested in. It is two-dimensional and has rows and columns.

Each entity type in the ER model is mapped into a relation.

* The attributes become the columns.
* The individual entities become the rows.

**Mapping of regular entity types**

Create relations and attributes for regular entity types. For each regular entity type E in the ER schema, create a relation R that includes all the simple attributes of E. Include only the simple component attributes of a composite attribute. Choose one of the key attributes of E as primary key for R. If the chosen key of E is composite, the set of simple attributes that form it will together form the primary key of R.

For example, we create the relations EMPLOYEE, DEPARTMENT, and PROJECT for the regular entity types EMPLOYEE, DEPARTMENT, and PROJECT in COMPANY ER schema. Then, we choose SSN, DNUMBER and PNUMBER as primary keys for the relations EMPLOYEE, DEPARTMENT, and PROJECT, respectively. Foreign keys are not included yet.

**Mapping of weak entity type**

For each weak entity type W in the ER schema with owner entity type E, create a relation R, and include all simple attributes of W as attributes of R. Include as a foreign key attributes of R the primary key attribute(s) of the relation(s) that correspond to the owner entity type(s). The primary key of R is the combination of the primary key(s) of the owner(s) and the partial key of the weak entity type W.

**For example,**

We create the relation DEPENDENT in this step. Include the primary key SSN of the EMPLOYEE as a foreign key attribute of DEPENDENT. The primary key of DEPENDENT is the combination {ESSN, DEPENDENT\_NAME} because DEPENDENT\_NAME is the partial key of DEPENDENT.

**Mapping of binary 1:1 relationship type**

For each binary 1:1 relationship type R in the ER schema, identify the relations S and T that correspond to the entity types participating in R. Choose one of the relations-S, say- and include as foreign key in S the primary key of T. It is better to choose as entity type with total participation in R. Include all the simple attributes of the 1:1 relationship type R as attributes of S. When both participations are total, we may merge the two entity types and the relationship into a single relation.

**Example**

Let’s consider the relationship type MANAGES which is total. Include the primary key of EMPLOYEE as foreign key in the DEPARTMENT and rename it as MGRSSN.

**Mapping of 1: N relationship type**

For each 1: N relationship type R, identify the relation S that represents the participating entity type at the N-side of R. Include as foreign key in S the primary key of the relation T that represents the other entity type participating in R. Include any simple attributes of the 1:N relationship type as attributes of S.

**Example**

Let’s consider the 1: N relationships type WORKS\_FOR, and CONTROLS. For WORKS\_FOR, include the primary key DNUMBER of DEPARTMENT as foreign key in EMPLOYEE and call it DNO. For CONTROLS, include the primary key DNUMBER of DEPARTMENT as foreign key in PROJECT and call it DNUM.

**Mapping of M: N relationship type**

For each M: N relationship type R, create a new relation S to represent R. Include as foreign key attributes in S the primary keys of the relations that represent the participating entity types; their combination will form the primary key of S. Include any simple attributes of the M:N relationship type as attributes of S. Note: We cannot represent an M: N relationship type by a single foreign key attribute.

**Example**

Let consider the M: N relationship type WORKS\_ON. Include the primary keys of the PROJECT and EMPLOYEE relations as foreign keys in WORKS\_ON and rename them as PNO and ESSN, respectively. Include an attribute HOURS in WORKS\_ON.

**Mapping of Multi-valued attributes**

For each multi-valued attribute A, create a new relation R. Include an attribute corresponding to A, plus the primary key attribute K as a foreign key in R. The primary key of R is the combination of A and K.

**Example**

Let’s consider the attribute DLOCATION. Create a relation DEPT\_LOCATIONS. The primary key of DEPT\_LOCATIONS is the combination of {DNUMBER, DLOCATION}.