Database Design & Implementation

COMP1302 Coursework

Student Name: Duong Bui Dinh

Student Roll Number: GT00010

Instructor: Vung Pham Van

Contents

Introduction

Designing database is an important stage in software development. This document is a report after designing database for Web Online Diary. All stages of this designing are follow with the common designing stages of designing database: Conceptual Model, Logical Model with Normalization and Physical Model.

D1 Assumption & Business rules

There are some assumptions to make some business rules clearly.

1. Each Blogger can have more than one blog. Each blog must belong to one blogger.
2. Each Blogger can post entry on his/her blog and other's blog.
3. One blogger can have many bankcards, or credit cards. However, there is only one bankcard or credit card is used for payment.
4. Each Bank Card or Credit Card must belong to a blogger.
5. Each entry has to pay for a payment. Each payment has to belong to an Entry.
6. Each Payment has a price, the price can change at any time, at this time, the price is two $ per one payment for entry.
7. An Entry has many comments; every comment can also have comments on it. All comments have to be categorised.
8. One question can be answered by concerned blogger and other bloggers.
9. Each Answer must belong to a Blogger
10. One blog has unique blog title in entire database.

D2 Conceptual model

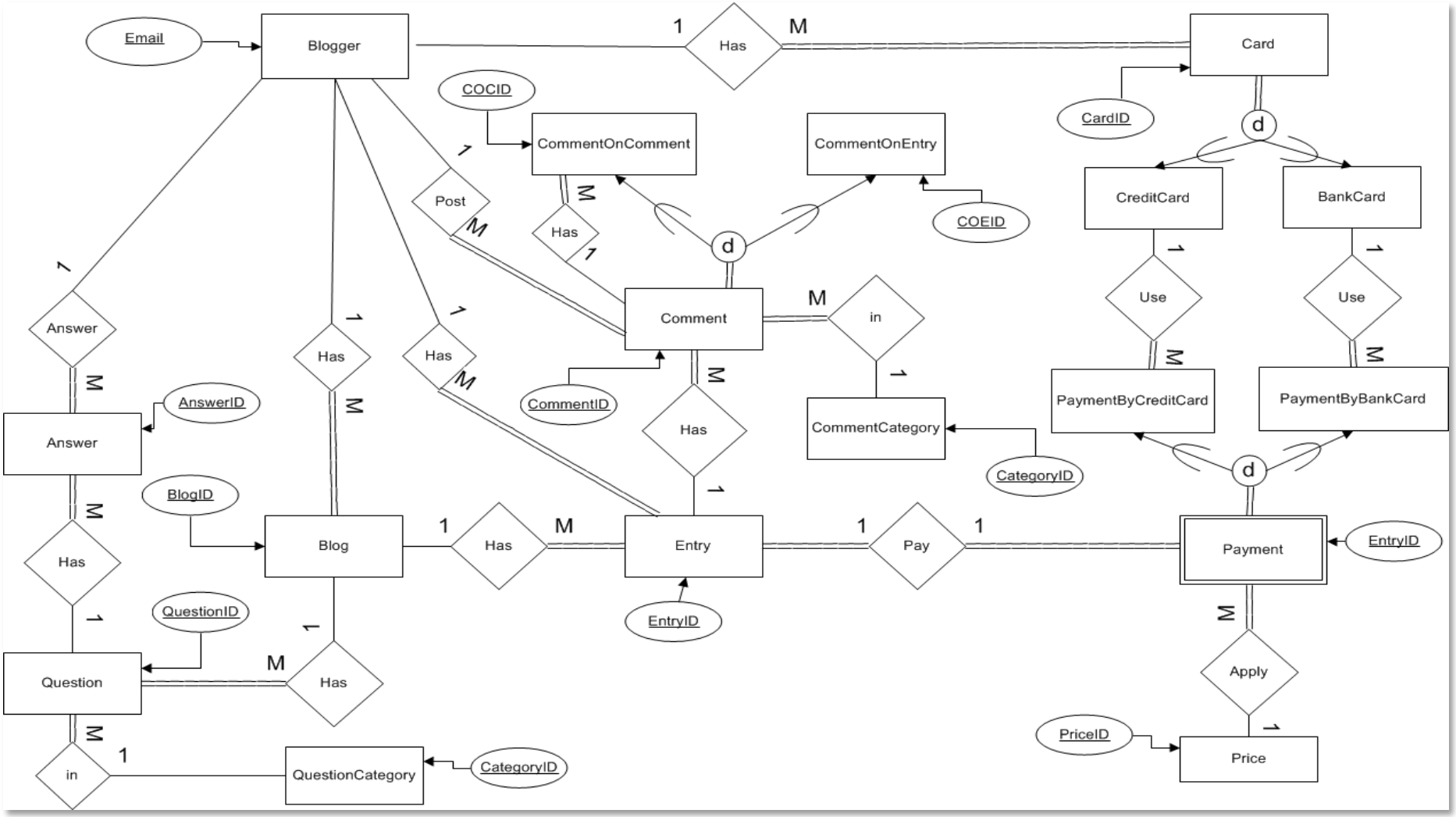

Figure 01: Conceptual model diagram (Chen's Notation)

This EER diagram shows data model of Web Online Diary. It contains Entities, relationships and primary key for each Entity. Attributes of Entities are show in Logical stage.

D3 Logical relation schema

This following Logical Relation is the result of mapping process from EERD above to relational schema.

It shows attributes of each entity in EERD above and links between relations (tables).

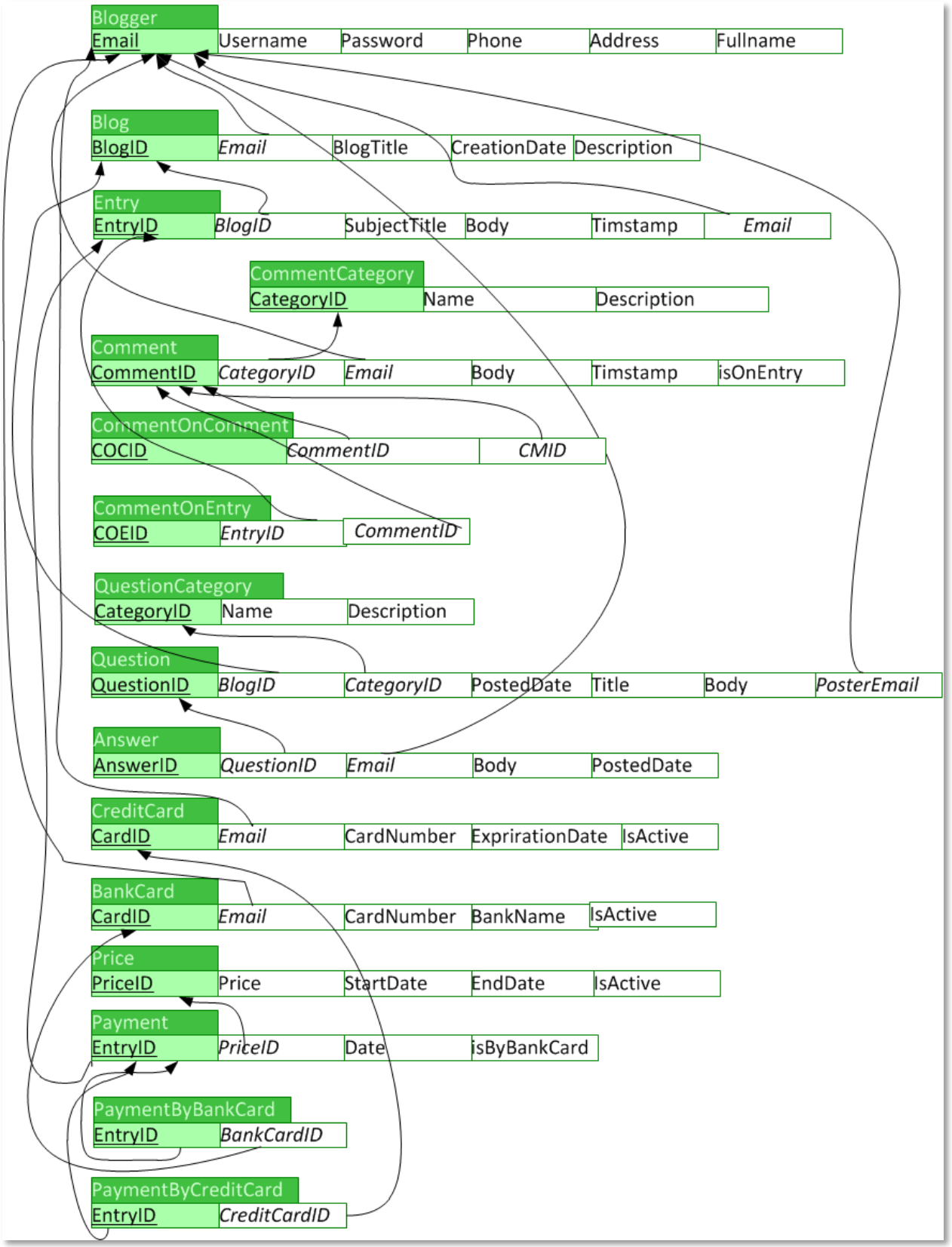

Figure 02: Relation schema shows links between relations

Database description

This is description for database.

Blogger			
Column	Data Type	Constraint	Description
Email	Text	Primary key	Identify blogger
Username	Text	Unique, Not null	Alias name
Password	Text	Not null	Password
Phone	Text	Not null	Phone
Address	Text	Not null	Address
Fullname	Text	Not null	Full name

Blog			
Column	Data type	Constraint	Description
BlogID	Number	Primary key	Indentify blogger
Email	Text	Foreign Key	References to **Blogger** table
CreationDate	Date/time	Not null	Creation date of blog
Description	Text	Not null	Description of blog
BlogTitlte	Text	Not null, unique	Title of blog

Entry			
Column	Data Type	Constraint	Description
EntryID	Number	Primary key, identity	Identify an entry
Email	Text	Foreign key	References to **Blogger** table
BlogID	Number	Foreign key	References **Blog** table
SubjectTitle	Text	Not null	Title of entry
Body	Text	Not null	Content of entry
Timestamp	Date/time	Not null	Posted time

CommentCategory			
Column	Data type	Constraint	Description
CategoryID	Number	Primary key, identity	Indentify a category
Name	Text	Not null	Name of category
Description	Text	Not null	Description of category

Comment			
Column	Data type	Constraint	Description
CommentID	Number	Primary key	Primary key
CategoryID	Number	Foreign key	References to **CommentCategory** table
Body	Text	Not null	Content of comment
Timestamp	Date/time	Not null	Posted time
Email	Text	Foreign key	Reference to **Blogger** table
isOnEntry	Yes/No	Not null	Determine what kind of comment

CommentOnEntry			
Column	Data type	Constraint	Description
COEID	Number	Primary key	
CommentID	Number	Foreign key	Reference to **Comment** Table
EntryID	Number	Foreign key	References to **Comment** table

CommentOnComment			
Column	Data type	Constraint	Description
COCID	Number	Primary key	
CommentID	Number	Foreign key	Reference to **Comment** table
CMID	Number	Foreign key	References to **Comment** table

QuestionCategory			
Column	Data type	Constraint	Description
CategoryID	Number	Primary key, identity	Primary key
Name	Text	Not null	Name of category
Description	Text	Not null	Description of category

Question			
Column	Data Type	Constraint	Description
QuestionID	Number	Primary key, identity	Primary key
BlogID	Number	Foreign key	References to **Blog** table
CategoryID	Number	Foreign key	References to **Category** table
PostedDate	Date/time	Not null	Posted time of question
Title	Text	Not null	Title of question
Body	Text	Not null	Content of question
PosterEmail	Text	Foreign key	Indentify the person who post this question, references to **Blogger** table

Answer			
Column	Data type	Constraint	Description
AnswerID	Number	Primary key	Primary key
QuestionID	Number	Foreign key	References to **Question** table
Body	Text	Not null	Content of answer
PostedDate	Date/time	Not null	Posted time of answer
Email	Text	Foreign key	References to **Blogger** table

CreditCard			
Column	Data type	Constraints	Description
CardID	Number	Primary key	Primary key
Email	Text	Foreign key	References to **Blogger** table
CardNumber	Text	Not null	Serial number of credit card
ExpirationDate	Date/time	Not null	Expiration Date
isActive	Yes/No	Not null	Determine if the card is using

BankCard			
Column	Data type	Constraint	Description
CardID	Number	Primary key	Primary key
Email	Text	Foreign key	References to

			Blogger table
CardNumber	Text	Not null	Serial number of bank card
BankName	Text	Not null	Name of bank
IsActive	Yes/No	Not null	The card is using

Price			
Column	Data type	Constraint	Description
PriceID	Number	Primary key	Primary key
Price	Number	Not null	Price of payment for each entry
StartDate	Date/time	Not null	Date are being to apply the price for payment
EndDate	Date/time	Allow null	End date of price
IsActive	Yes/No	Not null	Determine what price are using

Payment			
Column	Data type	Constraint	Description
EntryID	Number	Primary key, foreign key	Reference to **Entry** table
PriceID	Number	Foreign key	References to **Price** Table
Date	Date/time	Not null	Date of payment
IsByBankCard	Yes/no	Not null	Determine type of payment

PaymentByCreditCard			
Column	Data type	Constraint	Description
EntryID	Number	Primary key, foreign key	Reference to **Payment** table
CreditCardID	Number	Foreign key	References to **CreditCard** table

PaymentByBankCard			
Column	Data type	Constraints	Description
EntryID	Number	Primary key, foreign key	References to **Payment** table
BankCardID	Number	Foreign key	Reference to **BankCard** table

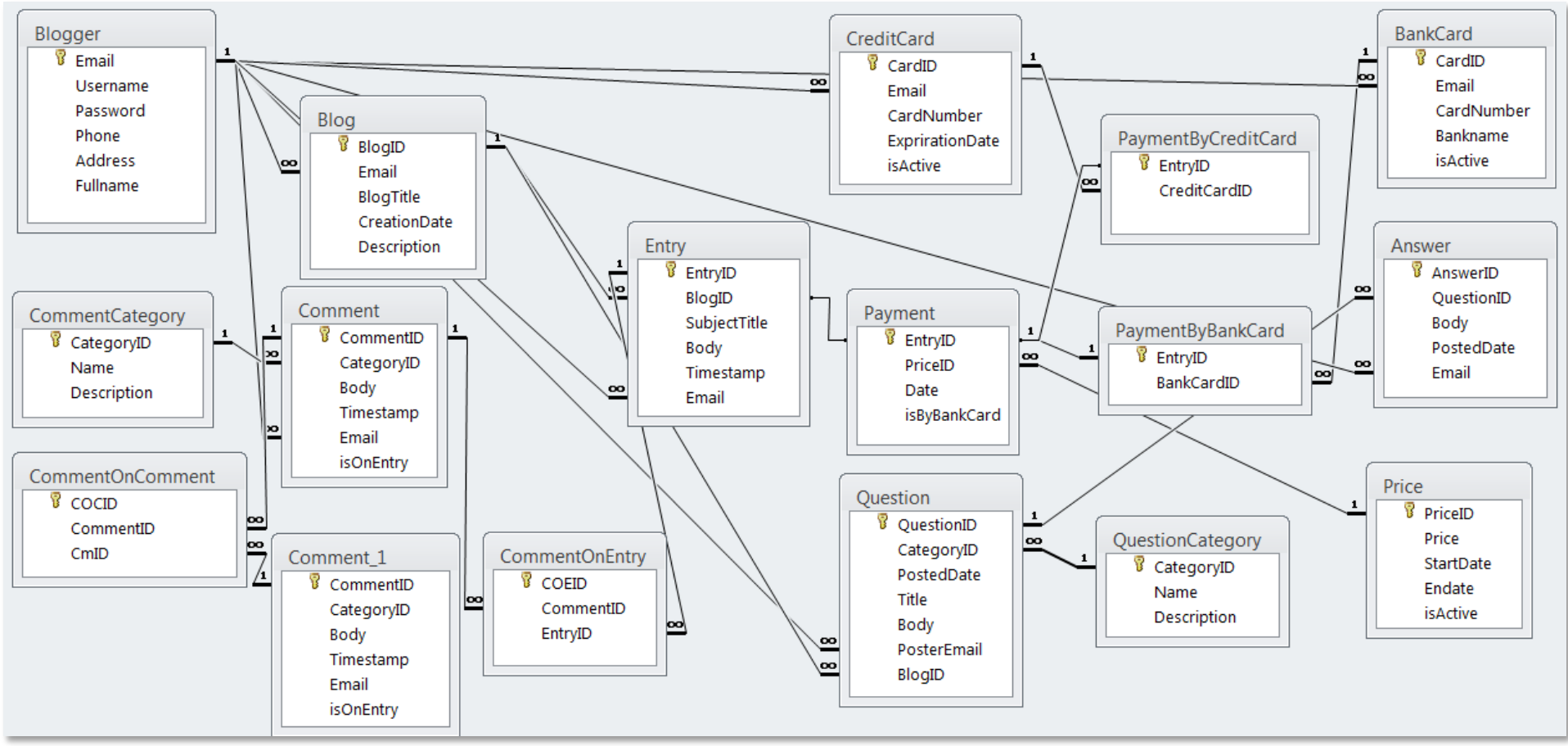

Figure 03: Database relationship

Note: the table Comment_1 in figure 03 is actual the table Comment. Column CmID and COCID in table CommentOnComment references to CommentID in table Comment together. But Access Database Relationship does not support reference like this. So it automatically generates this Comment_1 table to show relationship.

D4 Normalization check

"The Relational Schema satisfies 3NF criteria"

D5 Create Database

Database has been created in MS Access.

D6 SQL Code

A1

"List the titles and creation dates of all Blogs, as well as the total number of Blog entry associated with that blog" (Coursework requirement)

We can deal this requirement by using subquery, and **Count**() function.

```
SELECT b.BlogTitle, b.CreationDate, (SELECT COUNT(*) FROM Entry e WHERE
e.BlogID=b.BlogID) AS TotalEntry FROM Blog AS b;
```

BlogTitle	CreationDate	TotalEntry
Blog One	10/14/2011	4
Blog of Duong 2	10/5/2011	0
Blog of hack.wan	10/6/2011	1
Blog 2 of dinhduong.bui@gmail.com	10/12/2011	1

Figure 04: The result of A1 Query

A2

"Provide a list of all Bloggers (i.e. emails) who created entries on a given calendar month along with the total amount that each Blogger paid during that month." (Coursework requirement)

There are some steps to deal with this requirement:

1. **INNER JOIN** all tables related each other's: **Blogger, Entry, Payment, Price**
2. Filter only rows suitable with condition (in a given month- input value).
3. Using **Sum()** function for **Price** column and **Group By** blogger email

```
SELECT bl.Email, bl.Fullname, Sum(pr.price) AS Amount

FROM Price AS pr INNER JOIN (Blogger AS bl INNER JOIN (Entry AS e INNER JOIN Payment AS
p ON e.EntryID = p.EntryID) ON bl.Email = e.Email) ON pr.PriceID = p.PriceID

WHERE (((Month([p].[Date]))=[Enter a month]))

GROUP BY bl.Email, bl.Fullname;
```

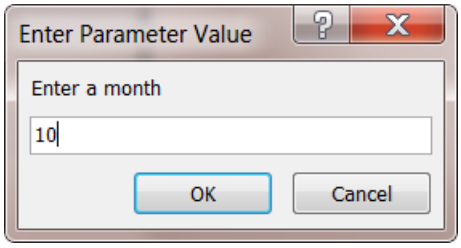

Figure 04a: Input form

Email	Fullname	Amount
dinhduong.bui@gmail.com	Duong	4
hack.wan@yahoo.com.vn	Bui Dieu Huong	2

Figure 04b: the result of A2 Query

A3

"Produce, for the first comment made on a given Blog entry, any further comments made on that original comment only. The user should be prompted for the Blog entry's unique identifier" (Coursework requirement)

There are some steps to deal this requirement:

1. Find the first **Comment** (get **CommentID)** on a given **Entry**
    a. **INNER JOIN Comment** and **CommentOnEntry** table, filter only rows with condition (given **EntryID)** then Using **Min**() function with **Timestamp** column to get the **Timestamp** of the **first Comment** for the given **Entry.**
    b. **INNER JOIN Comment** and **CommentOnEntry** table , get the fisrt **Comment** on **Entry** by comparing the condition is **Timestamp** above and filter by **EntryID** (supplied fisrt time) => got first **Comment** for given Entry, then **Select** only **CommentID**

2. Get all **CommentOnComment** for the first **CommentOnEntry**
   a. **INNER JOIN CommentOnComment** and **Comment** table
   b. Filter all **CommentOnComment** with condition is column **CMID** equals to **CommentID** above

```
SELECT Comment.CommentID,Body, Timestamp,Email

FROM Comment INNER JOIN CommentOnComment ON Comment.CommentID =
CommentOnComment.CommentID

WHERE CommentOnComment.CMID=

(SELECT CE.CommentID

FROM Comment  C INNER JOIN CommentOnEntry  CE ON C.CommentID = CE.CommentID

WHERE ((C.timestamp=(SELECT Min(timestamp)

FROM Comment AS C INNER JOIN CommentOnEntry AS CE ON C.CommentID = CE.CommentID

WHERE (((CE.EntryId)=[Enter EntryID])))));
```

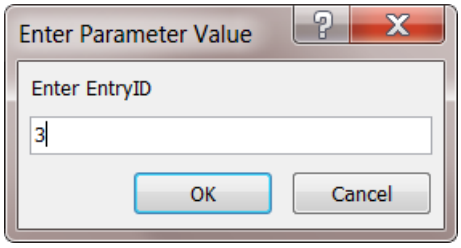

Figure 05a: Input entry ID

CommentID	Body	Timestamp	Email
4	Comment on Comment 3	10/11/2011	iamduong@yahoo.com
7	Comment for the comment 3 , Comn	10/29/2011	dinhduong.bui@gmail.com
8	Comment 3 is a comment for entry ic	10/29/2011	dinhduong.bui@gmail.com

Figure 05b: All the Comments comment on the first Comment of Entry has EntryID=3

A4

"Produce, on demand for a particular date range, a list of all Blog titles, creation date and their creators email addresses, The user should be prompted for the start and end date of the required period" (Coursework requirement)

There are some steps:

1. Get input value: **Start Date** and **End Date**

2. Select in Blog table what we need with comparing the condition

```
SELECT b.BlogTitle, b.CreationDate, b.Email

FROM Blog AS b

WHERE b.CreationDate>=[Start date] And b.CreationDate<=[End date];
```

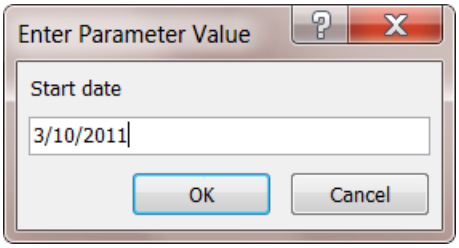

Figure 06a: Input start date

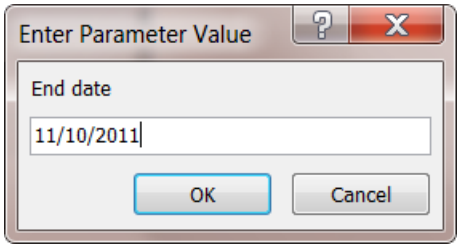

Figure 06b: input end date

	BlogTitle	CreationDate	Email
	Blog One	10/14/2011	dinhduong.bui@gmail.com
	Blog of Duong 2	10/5/2011	duongbdgt00010@fpt.edu.vn
	Blog of hack.wan	10/6/2011	hack.wan@yahoo.com.vn
	Blog 2 of dinhduong.bui@gmail.com	10/12/2011	dinhduong.bui@gmail.com

Figure 06c: the result of A4 query

D7 Registration Form

Figure 07: Registration Form

The form allow user to register to become a new Blogger.

When user fills full required data into the form, then click on **Register** button, this data will be inserted into Blogger table.

D8 A5-Form Report

"Given a particular Blogger list in a reverse order of a Blog entry (i.e. most recent entry first), the poster of the Blog entry and the creation time and date of the entry and the subject and the text of the entry for a particular Blog title." (Coursework requirement)

Figure 08: Chose a blogger to view report

When user selects a Blogger, the report will show all Entries of the Blogger. All Entries are grouped for each Blog

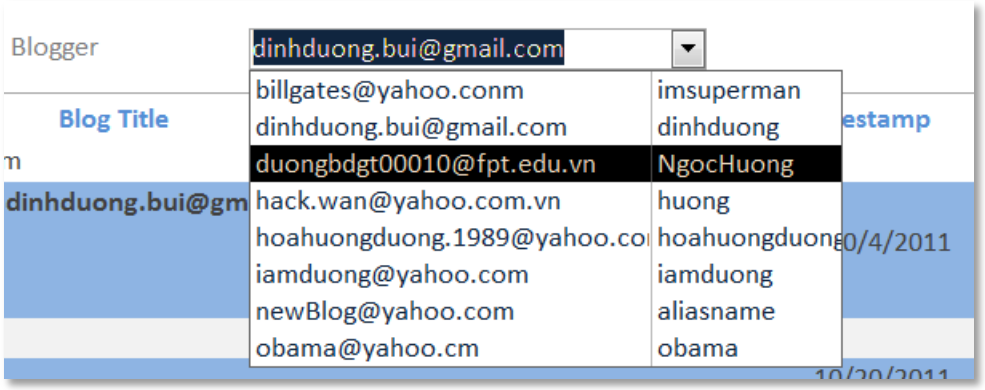

Figure 08b: Drop Down List shows all Bloggers

Figure 08b Description

When user clicks on the Combo Box, A Drop Down List will show all Bloggers. Then User can choose one of the Bloggers.

D9 A6-Master Detail Form

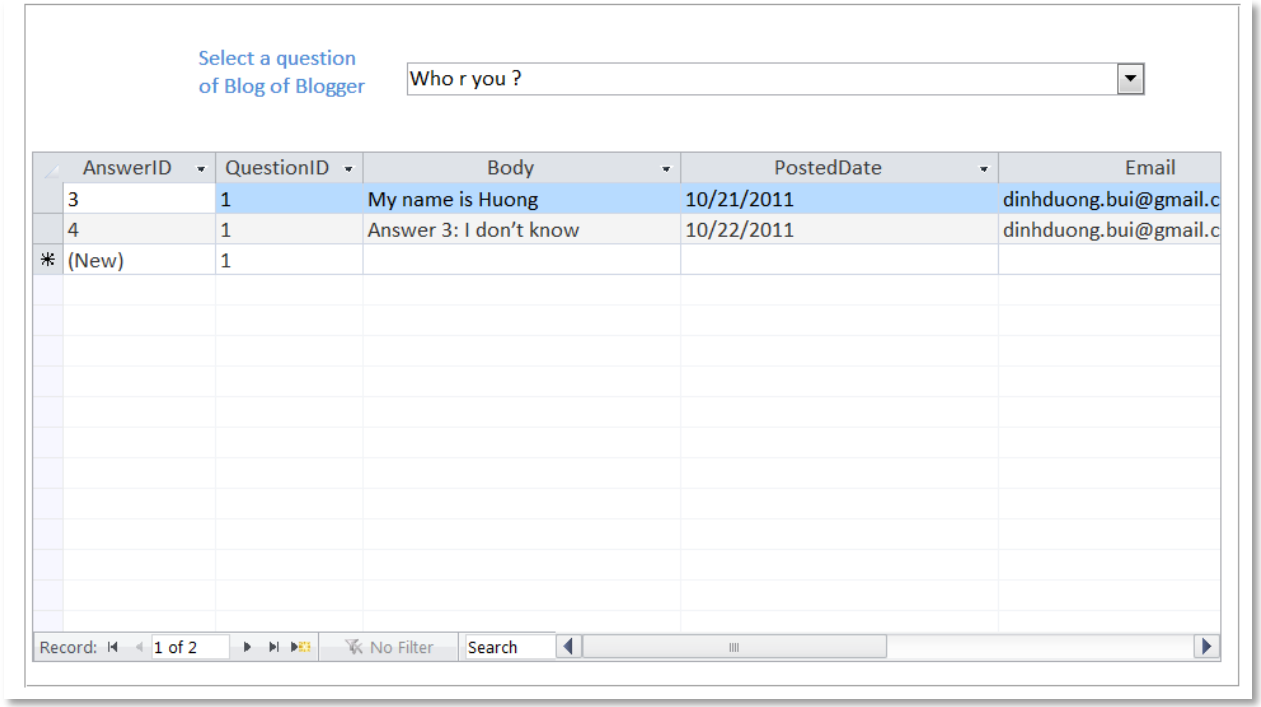

Figure 09: Master-Detail Form

Figure 09 Description

The master-detail form provides a list of all answers made on a given question for a given Blogger

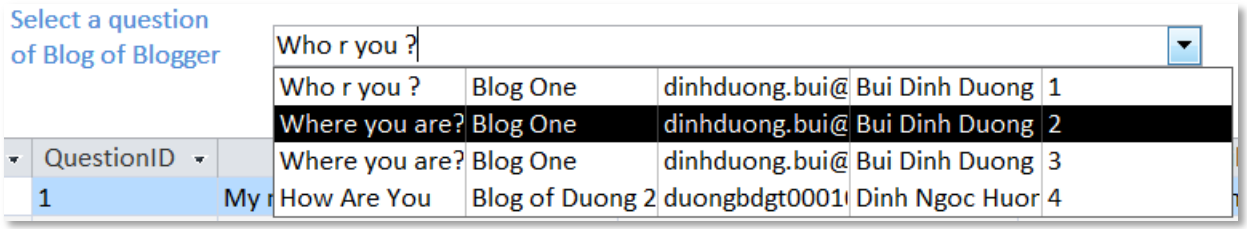

Figure 09b: Drop Down List for user selects a Question group by Blogger and Blog.

Figure 09b Description

When user clicks on the Combo box, A Drop Down List allow user to select a Question for a give Blogger and Blog

References

Coursework Requirement:

https://cms1.gre.ac.uk/collaborativeprogrammes/students/courseworks/coursework%202011-2012/Nov-Dec%202011/CW_COMP1302_190002_ver1_1011.pdf