Main Project
Database for Financial Institution

COMP 353
Databases

Presented to Khaled Jababo

Written by

Sotirios	
Akeel	
Andreas	
Abdullah	
Zaman	

Team ID: jmc353_1

Concordia University
Montreal
August 8th, 2011

DESIGN AND E/R MODEL

Below is our E/R Model:
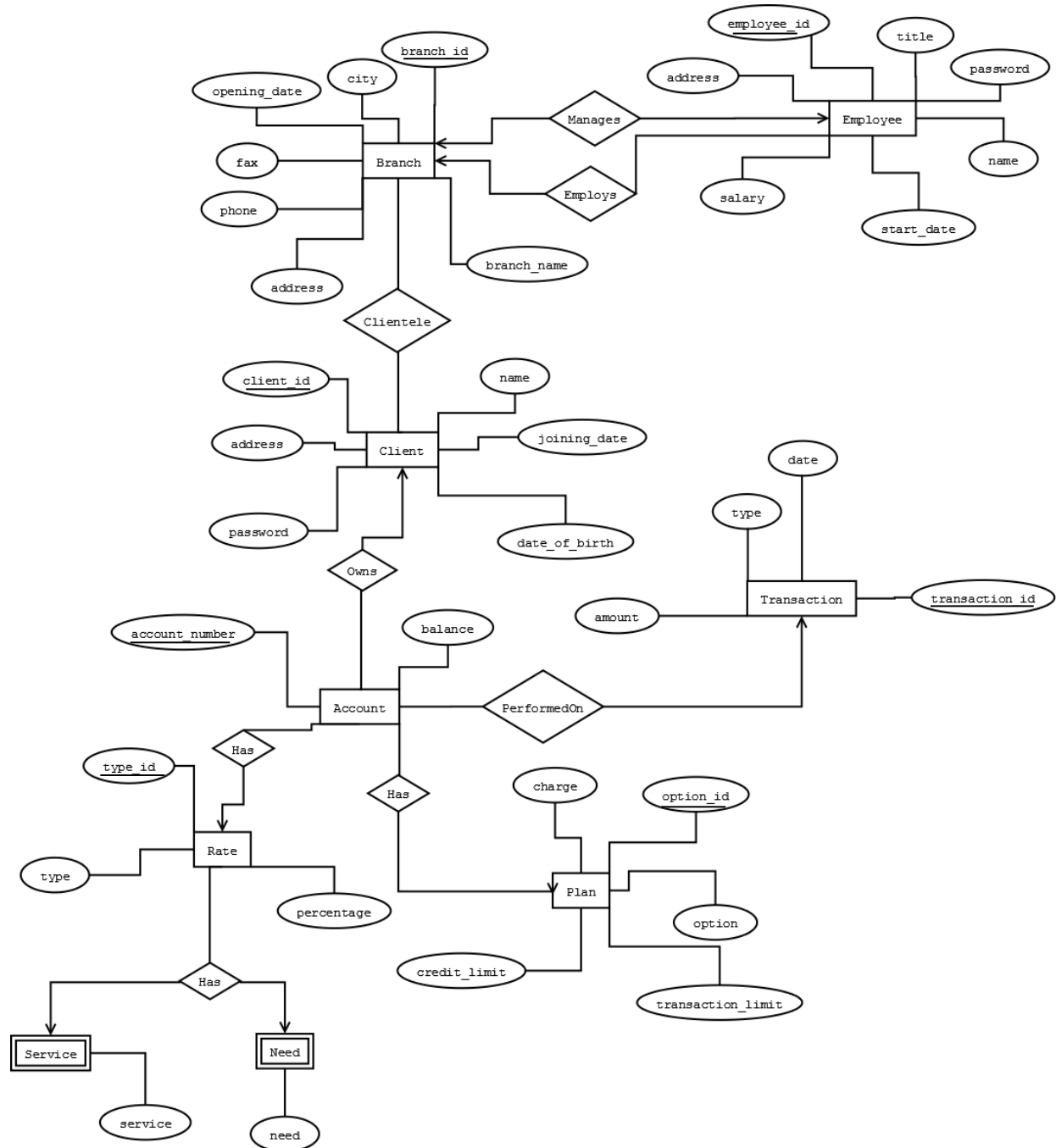
There are entities for each of the domain entities required: Branch, Client, Employee, Account, and Transaction. We added the entities Rate, Plan, Service, and Need to add conceptual meaning to the required entities.

CONVERSION INTO DATABASE SCHEMAS

Each (non-weak) entity has a defining id number as a primary key, ex. Branch has a branch_id.

In Branch, only the branch_id determines all the other fields.

The Manages relation is 1-to-1. We therefore added a field in the Branch table, manager_id, as a foreign key to the Employee table.

In Employee, only employee_id determines all the other fields. The Employs relation is 1-to-many. We added a field to Employee, branch_id, to represent which branch and employee works at. It is also a foreign_key.

The Clientele relation is many-to-many. We created a new table for it, where the client_id and the branch_id, combined, are the primary key.

In Client, only the client_id determines all the other fields. Because a client can have multiple accounts, the 'Has' relation is implemented as a field in Account for the client_id, as a foreign key. Similarly, the relation 'PerformedOn' between Account and Transaction is implemented as an account_number field in Transaction representing the account the transaction was performed on. For a transfer transaction, we have defined it as a withdrawal from one account and a deposit in another, so the single field is sufficient. The account_number and the transaction_id uniquely determine their entities' respective attributes.

The relations from Account to Rate and Plan are implemented as foreign keys to those tables through the type_id and option_id respectively.

The weak entities, Service and Need, although they do exist as tables in our database, simply appear as varchars in the Rate table.

All our relations are in 3NF since they all have one single field, their ID, that determines all their other fields.

You can find the actual structure of our database in the SQL queries included in the source code.

USER GUIDE

<u>Client Usage</u>

1. <u>Sign-up</u>: In order to create an account, you must visit a branch and have a teller or manager add you as a new client to the branch.
2. <u>Sign-in</u>: Once you've been added as a new client, you'll need to use your client_id and password (defaulted to 12345) to access your account.
3. <u>Operations:</u>
   a. <u>Branch:</u>
      i. <u>Read:</u> You can read information about all branches.
   b. <u>Client:</u>
      i. <u>Read</u>: You can read your client info (to update you need to visit a teller or manager)
   c. <u>Account:</u>
      i. <u>Read</u>: You can query information about the accounts that you own
   d. <u>Transaction:</u>
      i. <u>Withdraw</u>: Choose one of your accounts, and specify amount to withdraw
      ii. <u>Deposit</u>: Choose one of your accounts and amount to deposit
      iii. <u>Transfer</u>: Choose one of your accounts, specify amount, and the account to which you want to transfer money
      iv. <u>Read</u>: You can query information concerning all the transactions that were made to your accounts
4. <u>Other</u>: For other operations (eg. updating your client info), you will need to visit a branch and have a teller or a manager do that for you.

Figure 1.1 Login as Client

Figure 1.2 Operations as Client

<u>Teller (Employee) Usage</u>

1. <u>Account Creation</u>: To be added as an employee, you must have a manager create an account for you.
2. <u>Sign-in</u>: Once you've been added as a new employee, you'll need to use your employee_id and password (defaulted to 12345) to access your account.
3. <u>Operations:</u>
   a. <u>Branch:</u>
      i. <u>Read:</u> You can read information about all branches.
   b. <u>Client:</u>
      i. <u>Add:</u> You can add a new client.
      ii. <u>Update:</u> You can update any client's information.
      iii. <u>Read:</u> You can read any client info.
   c. <u>Employee:</u>
      i. <u>Read:</u> You can list the information of all employees of a certain branch.
   d. <u>Account:</u>
      i. <u>Create</u>: You can create an account and associate it with any client. Choose create from the dropdown menu, and select a client on the next page.
      ii. <u>Update</u>: You can update information associated with any account (ownership, balance, option, type).
      iii. <u>Read</u>: You can query information about any accounts.
   e. <u>Transaction:</u>
      i. <u>Withdraw</u>: Choose any account and specify an amount to withdraw.
      ii. <u>Deposit</u>: Choose any account and specify the amount to deposit.
      iii. <u>Transfer</u>: Choose any account to withdraw from specify amount and the account to which you want to transfer the money.
      iv. <u>Read</u>: You can query information concerning all the transactions that were made on any account.
4. <u>Other</u>: For other operations (eg. updating your employee info), you will need to talk to a manager.

Figure 2.1 Operations as Teller

JMC353_1 Bank

Login

Branch CRUD Actions

1. Read All Branch Information

Client CRUD Actions

1. [Add a new client]
2. [Update client information]
3. [Read client information]

Employee CRUD Actions

1. List Employees

Account CRUD Actions

1. Select Account Action
   [▼] [go]

Transaction CRUD Actions

1. Select Account & Specify Amount to Withdraw
   [Account 1, Owner: Colorado Wynn, Balance: 78373.0 ▼] Amount: [0] [Withdraw]
2. Select Account & Specify Amount to Deposit
   [Account 1, Owner: Colorado Wynn, Balance: 78373.0 ▼] Amount: [0] [Deposit]
3. Transfer Amount: [0]
   From: [Account 1, Owner: Colorado Wynn, Balance: 78373.0 ▼] To: [Account 1, Owner: Colorado Wynn, Balance: 78373.0 ▼]
   [Transfer]
4. Select Account to Query Transactions:
   [ALL ACCOUNTS ▼] [go]

Manager (Employee) Usage

A manager is defined by an employee having their employee_id stored in the manager_id of a branch.

As a manager, you have access to all CRUD (Create, Read, Update, Delete) operations of all entities (Branch, Employee, Client, Account, Transaction).

In other words, you can perform the same operations as a teller, with the following additions:

1. Extra Operations:
   a. Branch:
      i. Create: You can create a new branch.
      ii. Delete: You can delete a branch.
   b. Client:
      i. Delete: You can delete clients by specifying their ids.
   c. Employee:
      i. Create: You can create new employees by clicking on the "Insert Employee" link.
      ii. Update: You can create and update an employee's information by clicking on the appropriate link.
      iii. Delete: Use "Delete Employee" to delete a teller.

Figure 3.1 Operations as Manager

MEMBER CONTRIBUTIONS

Member Name	Contributions
Sotirios	- Team Manager - Branch CRUD implementation - UI and scripts - E/R model
Akeel	- Database design - Database implementation - Account and Transaction CRUD implementation - UI and scripts - User guide
Andreas	- Database design - Site interface
Abdullah	- Employee CRUD implementation - UI and scripts - Server manager - Testing
Zaman	- Client CRUD implementation - UI and scripts - E/R model

INSTALLATION GUIDE

For installation, the application requires a server with the following attributes:

- Php
- Mysql
- PhpMyadmin

The SQL queries for the database may be imported into phpmyadmin, the file for importation is attached in the project submission with the extension '.sql'.

The username and login reflect those stored in the database. Place the project files into the root directory of the server.

☐	📁 employee	--	8/7/2011 6:01:04 pm	Directory
☐	📁 header_files	--	8/7/2011 12:09:18 am	Directory
☐	📁 index_files	--	8/7/2011 1:14:44 am	Directory
☐	account.cgi	7.93 KB	8/7/2011 6:10:54 pm	Script
☐	addclient.php	2.74 KB	8/7/2011 6:00:18 pm	Script
☐	Client.html	472 B	8/7/2011 1:33:30 am	HTML File
☐	Client.php	472 B	8/7/2011 1:34:32 am	Script
☐	create_branch.html	1.19 KB	8/7/2011 1:33:31 am	HTML File
☐	create_branch.php	1.07 KB	8/7/2011 6:00:30 pm	Script
☐	create_employee.html	474 B	8/7/2011 1:33:34 am	HTML File
☐	db.php	501 B	8/7/2011 6:00:31 pm	Script
☐	db_info.php	171 B	8/7/2011 6:00:32 pm	Script
☐	delete_branch.html	686 B	8/7/2011 1:33:38 am	HTML File
☐	delete_branch.php	347 B	8/7/2011 6:00:33 pm	Script
☐	delete_employee.html	232 B	8/7/2011 1:33:39 am	HTML File
☐	deleteclient.php	2.45 KB	8/7/2011 6:00:32 pm	Script
☐	employee_index.html	424 B	8/7/2011 1:33:41 am	HTML File
☐	footer.html	3.42 KB	8/7/2011 12:05:13 am	HTML File
☐	footer.php	3.42 KB	8/7/2011 1:33:42 am	Script
☐	header.html	14.9 KB	8/7/2011 12:05:14 am	HTML File
☐	header.php	14.9 KB	8/7/2011 1:33:42 am	Script