

# INFO 6210

***Data Management and Database Design***

***Project topic Banking Management System***

## By

***Parul Chaudhary Sri Pooja Gade***

***Sujay Sathyanarayana Joshi Swamy Honnebagi Mallappa***

***Instructor***

***Wu-Ping Simon Wang Date: - 04/14/2019***

### Database Purpose:

The purpose of this database is to construct, record and maintain the transactions and services of the bank customers and employees. Further, this database can track and update all the data of bank customers and employees.

### Business Problem Addressed:

Banks have to deal with a lot of customers on a daily basis. The process of maintaining and monitoring each and every activity, exchanges related to each customer or employee on paper as records is an extremely cumbersome and a tedious process, that might result errors in some cases. Banks offer many benefitting services which attracts large number of customers. Due to increase in the number of customers, it is essential to upgrade from maintaining records on paper to maintain records in a database.

### Business Rules:

* An employee or a customer must have one or more bank accounts.
* An employee or a customer may have taken one or more loans.
* An employee or a customer may opt for one or more card services.
* An employee or a customer may have one or more insurances.
* An employee or a customer may opt for one or more alert services.
* An employee or a customer may opt for one or more reporting services.
* An employee or a customer with a bank account may access one or multiple bank branches.
* An employee or a customer must transact once or more within a specified time.

### Design Requirements (credit to professor Wu-Ping Simon Wang):

* Use Crow’s Foot Notation.
* Specify the primary key field in each table.
* Draw a line between the fields of each table to show the relationships between each table. The line should be pointed directly to the fields in each table that are used to form the relationship.
* Specify which is the one side of the relationship by placing a one next to the field where the line starts.
* Specify which table is on the many side of the relationship by placing a crow’s feet symbol next to the field where the line ends.

### Entity Relationship Diagram:

**A screenshot of a cell phone

Description automatically generated**

**Key Database Design Decisions:**

|  |  |  |
| --- | --- | --- |
| **Entity Name** | **Why entity included** | **How entity is related to other entities** |
| Employee | Employee entity is a | As a core entity in the |
|  | primary entity that contains | database, the primary key |
|  | all the details of employees | (i.e. emp\_id) of the |
|  | working in the bank. It is | employee entity is used to |
|  | essential to maintain an | determine the bank |
|  | employee entity because an | account, to which branch |
|  | employee is the one who | the employee belongs or |
|  | will deal with the | has taken any loan or |
|  | customers. | insurance. It is also used in |
|  |  | the associative entities |
|  |  | such as reporting services |
|  |  | has employee, alert service |
|  |  | has employee through |
|  |  | many to many |
|  |  | relationships. |

|  |  |  |
| --- | --- | --- |
| Customers | Customer entity is an essential entity that contains all the details of the customers | The primary key of the customer entity (i.e. cust\_id) is used to determine the bank account, what kind of card services they are using and what insurance or loan they have taken. A customer can have one to many relationships with the number of alerts they can receive, reporting services and number of card services. |
| Account\_Info | This entity includes all the details regarding the customer’s or employees accounts and the bank branch information. This entity cannot be a null entity. | This is one of the main entities where the bank account details of the employee and customers are stored. The Account\_Info entity is connected to employee and customers entity through many to many relationship in order to determine their account details respectively. And also the bank branch information of an employee and customer can be found as Account\_Info is connected to Bank\_Branch\_Info through branch\_id as a foreign key. |
| Bank\_Branch\_Info | This entity contains information regarding a particular bank branch such as branch id, name and address. | The Bank\_Branch\_Info entity is connected to Account\_Info entity in order to give the bank branch information of the employee and the customer through branch\_id as a foreign key in Account\_Info. Which in turn connected to employee and customer entities. |

|  |  |  |
| --- | --- | --- |
| Card\_Service\_Info | This entity stores the type of services provided by a card. | The Card\_Service\_Info entity is connected to employee and customers entity through many to many relationship in order to determine the employee and customer card services. |
| Employee\_has\_Card\_Service\_info | This entity stores the details related to the employees that work with the card services. | This entity acts like a bridge table between employee and  Card\_Service\_Info enitities. Where the primary keys of both the entities (i.e. emp\_id and card\_service\_id) are taken in order to determine the employee card services through many to many relationship. |
| Card\_Service\_info\_has\_Customers | This entity stores the details of a particular customer along with his loyalty points accessing a particular type of card service. | This entity acts like a bridge table between customers and Card\_Service\_Info enitities. Where the primary keys of both the entities (i.e. cust\_id and card\_service\_id) are taken in order to determine the customer card services through many to many relationship. |
| Loan | This entity includes all the details of the employee or customer who has taken the loan. | The loan entity is connected to employee and customers entity through many to many relationship in order to determine the type of the loan. |
| Employee\_has\_Loan | This entity includes the details of employee who has taken a loan. | This entity acts like a bridge table between employee and loan enitities. Where the primary keys of both the entities (i.e. emp\_id and |

|  |  |  |
| --- | --- | --- |
|  |  | loan\_type) are taken in order to determine whether an employee has a loan through many to many relationship. |
| Customers\_has\_Loan | This entity includes all the details of the customer who has taken a loan. | This entity acts like a bridge table between customers and loan enitities. Where the primary keys of both the entities (i.e. cust\_id and loan\_type) are taken in order to determine whether the customer has a loan through many to many relationship. |
| Insurance | The purpose of this entity is to maintain the data on insurances. | The Insurance entity is connected to employee and customers entity through many to many relationship in order to determine the insurance they have taken. |
| Employee\_has\_Insurance | This entity includes the details of the employees who have an insurance in this bank. | This entity acts like a bridge table between employee and Insurance enitities. Where the primary keys of both the entities (i.e. emp\_id and insurance\_id) are taken in order to determine whether an employee has an insurance through many to many relationship. |
| Customers\_has\_Insurance | This entity stores the data of the customers who have taken insurance in the bank. | This entity acts like a bridge table between customers and Insurance enitities. Where the primary keys of both the entities (i.e. cust\_id and insurance\_id) are taken in order to determine whether the customer has an insurance through many to many relationship. |

|  |  |  |
| --- | --- | --- |
| Alert\_Services | This entity includes the details regarding the charges associated with each alert. | The Alert\_Services entity is connected to employee and customers entity through many to many relationship in order to determine the alert services (i.e. phone or email) they have opted. |
| Alert\_Services\_has\_Employees | This entity stores the | This entity acts like a |
|  | information of the alert | bridge table between |
|  | services ( phone or email ) | employee and |
|  | opted by the employees of | Alert\_services enitities. |
|  | the bank. | Where the primary keys of |
|  |  | both the entities (i.e. |
|  |  | emp\_id and service\_id) are |
|  |  | taken in order to determine |
|  |  | the employee alert service |
|  |  | through many to many |
|  |  | relationship. |
| Alert\_Services\_has\_Customers | This entity includes the data | This entity acts like a |
|  | of the alert services (phone | bridge table between |
|  | or email) associated with | customers and |
|  | the customer. | Alert\_services enitities. |
|  |  | Where the primary keys of |
|  |  | both the entities (i.e. |
|  |  | cust\_id and service\_id) are |
|  |  | taken in order to determine |
|  |  | the customer alert service |
|  |  | through many to many |
|  |  | relationship. |
| Reporting\_Services | This entity contains the data | The Reporting\_Services |
|  | associated with making | entity is connected to |
|  | reports related to a | employee and customers |
|  | particular customer or | entity through many to |
|  | employee activity. | many relationship in order |
|  |  | to determine their |
|  |  | reporting services. |
| Reporting\_Services\_has\_Employee | This entity contains data | This entity acts like a |
|  | related to report an | bridge table between |
|  | employee activity. | employee and |
|  |  | Reporting\_services |
|  |  | enitities. Where the |
|  |  | primary keys of both the |
|  |  | entities (i.e. emp\_id and |

|  |  |  |
| --- | --- | --- |
|  |  | report\_id) are taken in order to determine the employee reporting service through many to many relationship. |
| Reporting\_Services\_has\_Customers | This entity contains data | This entity acts like a |
|  | related to report a customer | bridge table between |
|  | activity. | customers and |
|  |  | Reporting\_services |
|  |  | enitities. Where the |
|  |  | primary keys of both the |
|  |  | entities (i.e. cust\_id and |
|  |  | report\_id) are taken in |
|  |  | order to determine the |
|  |  | customer reporting service |
|  |  | through many to many |
|  |  | relationship. |
| Transactions | This entity contains all the | The is a final and most |
|  | details related to any | important entity where the |
|  | transaction that has been | transaction amount and |
|  | carried out. | the type is stored. |
|  |  | Transactions entity is |
|  |  | connected to Account\_info |
|  |  | entity through 1 to many |
|  |  | relationship. This |
|  |  | determines the transaction |
|  |  | of an employee and |
|  |  | customer through |
|  |  | account\_number (foreign |
|  |  | key in Transaction entity). |