REPORT

on

TELECOMMUNICATION DATABASE MANAGEMENT FINAL PROJECT

Prepared for

MSc. Nguyễn Thanh Tuấn

Instructor of Database Management System class

Faculty of Mathematics Economics

National Economics University

# INTRODUCTION

## Main Idea

This is the report of the final project for Database Design from group 2, we designed a database to manage a telecommunication company. The company has many branches, each branch would have a store. Each store would have many employees. Customer when signs up would have a contract, each contract goes with a service, each service could sign up for many times. There are two types of customers, prepaid and postpaid customers. Each prepaid customer has a customer activities table to track what they did, and also an account balance to track. Each postpaid customer has a receipt. There would be employees collect money every month on the 5th,15th,25th. Customers can comment and rate the service on a 5-star scale. While using the services, if customers have any troubles, they can report the error and there will be employee support and help them fix the errors. There is also a logger to log changes.

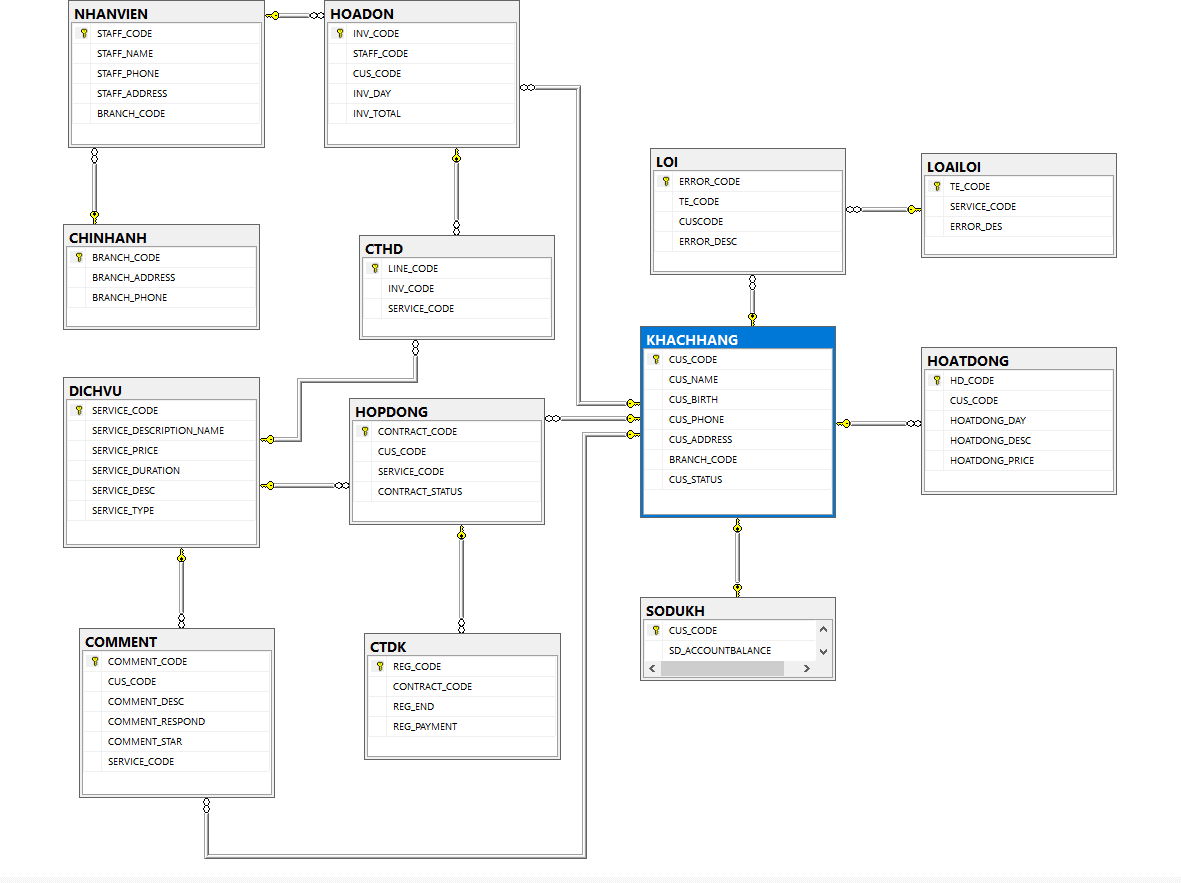
## Entities

* Branch(Chi nhánh)
* Staffs (Nhân viên)
* Customers (Khách hàng)
* Contract (Hợp đồng)
* Signing details (Chi tiết đăng ký - CTDK)
* Customer’s Activities (Hoạt động khách hàng)
* Customer’s Balance (Số dư khách hàng)
* Receipt (Hóa đơn)
* Receipt details (Chi tiết hóa đơn CTHD)
* Services (Dịch vụ)
* Comment (Bình luận)
* Error types(Loại Lỗi)
* Errors(Lỗi)

# ERD

## Graph

The Diagram below is our group ERD, following by their relationships



Relationships Table

|  |  |  |
| --- | --- | --- |
| Entity 1 | Entity 2 | Relationship |
| CHINHANH | NHANVIEN | 1:M |
| CHINHANH | KHACHHANG | 1:M |
| NHANVIEN | HOADON | 1:M |
| HOADON | CTHD | 1:M |
| DICHVU | CTHD | 1:M |
| DICHVU | HOPDONG | 1:M |
| DICHVU | COMMENT | 1:M |
| HOPDONG | CTDK | 1:M |
| KHACHHANG | HOADON | 1:M |
| KHACHHANG | HOPDONG | 1:M |
| KHACHHANG | COMMENT | 1:M |
| KHACHHANG | LOI | 1:M |
| KHACHHANG | SDKH | 1:1 |
| KHACHHANG | HOATDONG | 1:M |
| LOAILOI | LOI | 1:M |

# Meaning of tables and what do they do?

## Branch (Chi nhánh)

### Functions:

This telecommunication is national-wide, therefore it would have to manage a lot of regions. Each region is managed by a branch, and also, each branch would manage all customers that have the same BRANCH\_CODE.

### Primary key:

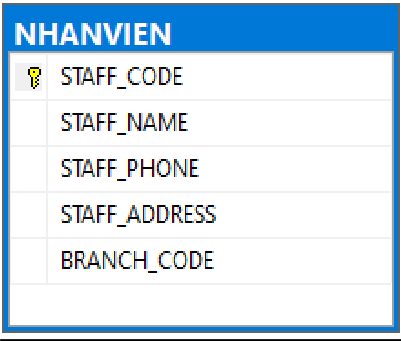
BRANCH\_CODE: each branch would have a single unique code to differentiate it from another branch.

### Foreign key(s): NONE Others:

BRANCH\_ADDRESS: Store the branch address so customers can work directly with the branch when they are in need.

BRANCH\_PHONE: Store the branch phone number so customers can call the branch when they are in need.

## Staffs (Nhân viên)



### Functions:

The table NHANVIEN manages all employees that work for the company, each staff would have a BRANCH\_CODE, which indicates the branch and also the region that they are responsible for.

### Primary key:

STAFF\_CODE: each staff would have a unique code to distinguish them from each other.

### Foreign key(s):

BRANCH\_CODE: references from CHINHANH, to know which branch they work for.

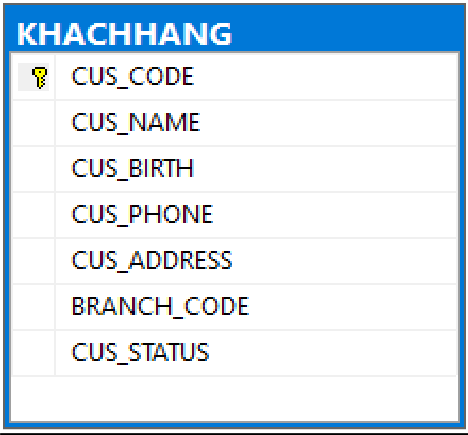
### Others:

STAFF\_NAME: *Store staff’s name*. STAFF\_PHONE*: Store staff’s phone to*

contact them.

STAFF\_ADDRESS: *Know where they live.*

## Customers (Khách hàng)



### Functions:

This table manages all customers of the company, it stores important information of customers.

### Primary key:

CUS\_CODE: unique code to distinguish user.

### Foreign key(s): NONE

**Others**:

CUS\_NAME: *Stores customer’s name.*

CUS\_BIRTH: Stores customer’s birthday to send them presents on their birthday.

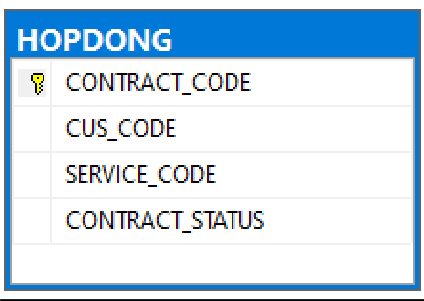
CUS\_PHONE: *Store customer’s number.*

CUS\_ADDRESS: *Know exactly where they live.*

BRANCH\_CODE: *Know the area that they live in.*

CUS\_STATUS: Check if they are a normal or blacklisted customer. A blacklisted customer is a postpaid customer who fails to pay money two consecutive times.

## Contract (Hợp đồng)



### Functions:

Each customer can sign up for many services so that we created this table to store all customer’s contracts, each contract contains information of a(1) customer and a (1) service that customer uses, if a customer decides to use another service, a new contract would be created to store the needed information.

### Primary key:

CONTRACT\_CODE, *a unique code to distinguish customers.*

### Foreign key(s):

CUS\_CODE from KHACHHANG, know which customer that contract belongs to.

SERVICE\_CODE *from DICHVU, know what service that customer uses.*

### Others:

CONTRACT\_STATUS: indicates the current status of a contract, if a user is using that contract, the status will be active, else it will be inactive.

## Signing details (Chi tiết đăng ký)

### Functions:

Customers can sign up for service multiple times so that we created this table to track the signing details of customers. For example, Customer A can use service SMS30 three times, every time customer A uses the service, there will be a new signing detail.

### Primary key:

REG\_CODE**,** *unique code to differentiate customers.*

### Foreign key(s):

CONTRACT\_CODE references from HOPDONG, to know which contract the details belong to.

### Others:

REG\_END: check when the customer has to pay money.

REG\_PAYMENT: check if the customer paid or pending to pay money, if a customer fails to pay money 2 times in a row, they would be blacklisted.

### Customer’s Activities (Hoạt động khách hàng)

### Functions:

Track all activities that a customer makes therefore can know which customer pays the most money so that the company can give them vouchers, coupons, etc...

### Primary key:

HD\_CODE: unique code to differentiate others code.

### Foreign key:

CUS\_CODE**:** Indicate which customer the activities belong to.

### Others:

HOATDONG\_DAY: *Records which day that activity was made.*

HOATDONG\_DESC**:** *Records what was the activity, or activity descriptions*.

HOATDONG\_PRICE**:** *Records the change in money. Could be positive or negative.*

## Customer’s Balance (Số dư khách hàng - SDKH)

### Functions:

See the Customer’s balance.

### Primary key:

CUS\_CODE**:** See which customer the balance belongs to. CUS\_CODE is unique.

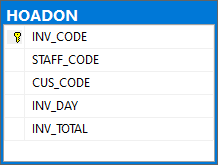
### Foreign key(s): NONE

**Others:**

SD\_ACCOUNTBALANCE: The number of money that customer has in float

12

## Receipt (Hóa đơn)

**Functions**: *See the total number of money that a customer has to pay after a period. The money is the sum of the price of all services that the customer used.*

**Primary key**:

INV\_CODE: *unique code to distinguish it from other codes.*

**Foreign key(s):**

STAFF\_CODE: *See which employee is responsible for collecting money*.

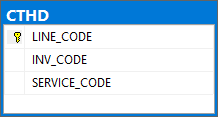
CUS\_CODE: *Which customer pays money.*

**Others:**

INV\_DAY: *The time that the money is collected.*

INV\_TOTAL: *The total money that customer paid.*

## Receipt details (Chi tiết hóa đơn CTHD)

**Functions**: *Each receipt has many small services that lie in it, this table helps give the company to see the receipt clearer.*

**Primary key**:

LINE\_CODE: *unique code to distinguish it from other codes.*

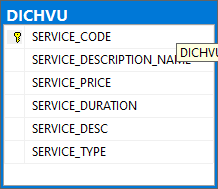
**Foreign key(s):**

INV\_CODE: *references from HOADON. See which receipt the receipt details belongs to.*

SERVICE\_CODE: *references from DICHVU. See which service has been used.*

**Others: NONE**

## Services (Dịch vụ)



### Functions:

Stores a list of all services and all related information.

### Primary key:

SERVICE\_CODE**:** *Unique code to distinguish it from other codes.*

### Foreign key(s): NONE Others:

SERVICE\_DESCRIPTION\_NAME: *Stores*

the name for the service.

SERVICE\_PRICE: *The price of the service.*

SERVICE\_DURATION: *The duration of the service, how long will it last.*

SERVICE\_DESC: *What users get when they sign up for the service.*

SERVICE\_TYPE: *Indicate whether the service is prepaid or postpaid.*

## Comment (Bình luận)

### Functions:

Let users leave comments for the services, employee can read those comments to further improve the services. There is also a star rating so that employees can quickly see which service has the lowest or highest rating so that employees can respond accordingly.

### Primary key:

COMMENT\_CODE: *the unique code to distinguish it from other codes.*

### Foreign key(s):

CUS\_CODE: references from KHACHHANG, see which user posted the comment.

SERVICE\_CODE: references from DICHVU, see which services the customer is referring to.

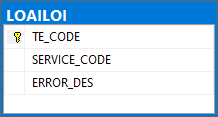
### Others:

COMMENT\_DESC: The message that customers want to leave there for the service.

COMMENT\_RESPOND: *The response from the employee, represents the company.*

COMMENT\_STAR: The star on a scale of 5 that customer wants to rate the service, from this scale, the company can know top services that have the best customer satisfaction.

## Error types(Loại Lỗi)



**Functions**:

A list of errors so that customers can easily choose when they want to report which error they have encountered, without retyping it to the question box, employees at the company could also diagnose the error and then fix it as soon as possible.

### Primary key:

**TE\_CODE**: Unique code to differentiate from others.

### Foreign key(s): NONE

**Others:**

SERVICE\_CODE: *Indicate the service where the error occurred.*

**ERROR\_DESC**: *Error Description, when*

and how,….

## Errors(Lỗi)

### Functions:

Stores all errors that customers reported. Employees can read to fix and improve services.

### Primary key:

ERROR\_CODE: *Unique code to differentiate with others.*

### Foreign key(s):

TE\_CODE: References from LOAILOI, uses to classify the error so that the employee can support customers as quickly as possible.

CUS\_CODE: References from KHACHHANG. Indicate which customer reported the error.

### Others:

ERROR\_DESC: *Further information for the error.*

# BUSINESS RULES

## Query

|  |  |  |
| --- | --- | --- |
| No. | Query | Meaning |
| 1 | What services customer A are using | Customer A can see which services they are using to serve their needs. |
| 2 | Total revenue from the prepaid customers. | See the revenue and compare it to postpaid revenue to have a better business strategy |
| 3 | Total revenue from the postpaid customers. | Compare it with prepaid revenue as mentioned above |
| 4 | See which service has been signed up the most | See services that have high demand, therefore the company can have a better business strategy such as creating similar services in the future or focus on advertising those services. |
| 5 | See the number of customer in a region/area | Can add or remove employees to serve the customers better. |
| 6 | See customer's responses base on the 3-star-and-up rated service. | See response from customers, improve or fix the services accordingly |
| 7 | Query the customers that have a birthday in that month | Wish them a happy birthday!!! |
| 8 | Check who collects and the amount of money from the customer. | When customers want to know more about their money, they can query from the system, who collected and also the amount of collected money |
| 9 | Which customers used all SMS services | Offer promotions for customers and introduce them to another service |
| 10 | Which customer comment the most | The customers who comment means that they do pay attention to the company’s services, the company should focus on their opinions. |
| 11 | Who paid the most for calling services. | Offer them better-priced services |
| 12 | Which services have the least amount of people use/sign up for. | See the reasons, fix it or stop running those services. |
| 13 | What services customer A sign up for in day ddmmyyyy. | Manage customer activities. |

## View

|  |  |  |
| --- | --- | --- |
| No. | View | Reason |
| 1 | View top 10 customers that paid the most money | Offer them coupons or vouchers as a part of the promotion program. |
| 2 | View sorted list of services by the number of users | Know customer’s demand, therefore can plan a better marketing strategy. |
| 3 | View customers who are on the blacklist. | When a customer can not fulfill the agreement, they will be listed on the blacklist, customers who are on the blacklist will not be able to sign up for another service, by creating this view, the employee can quickly search for blacklisted customers. |
| 4 | See postpaid customers | Easier to manage customers. |
| 5 | See prepaid customers | Easier to manage customers. |

## Procedure

|  |  |  |
| --- | --- | --- |
| No. | Procedure | Meaning |
| 1 | Every month on 5th 15th 25th, the company will calculate the money that they have to collect from those customers who signed in during the period accordingly. View who has to pay.  If the customers are not able to pay the money within 20 days, they will be moved to the blacklist. | The procedure helps manage the collecting and money flow process. |
| 2 | A procedure that checks if customer A has used the services or not using customer ID and service ID as parameters | The procedure helps the customers satisfies their needs |
| 3 | When a customer reports an error, check if that was spam or not. The procedure will check if a customer used the service or not, if not, it would classify the report as spam. | Since people could freely report errors, some of them might have not used the service yet, so this procedure helps screening the reports. |
| 4 | Check if the customer is late. If they are late 10 days, update the status to ‘late 10 days’, if they are 20 days late, update the status to ‘late 20 days’ and set the CONTRACT\_STATUS in  the contract to ‘inactive’. | Update the current status in each table. |
| 5 | Check if any customers have expired services today. | Update and manage customers. |

## Trigger

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Function | What it does | Meaning |
| 1 | Customer sign up for new service | When a customer signs up for a new service, check if whether or not they are on the blacklist.   * If yes, they will not be able to sign up. * If no => check if that customer has ever signed up for that service.   + If they have signed up before and the current status is not “active”=> Update the signing details status to:” Active”.   + If the status is already “Active”, add and update the date in the table Signing details (CTDK).   If they have not signed up then create a new contract and then enter all the signing information and update the tables accordingly | Perform all the necessary tasks when a new customer joins the network |
| 2 | Customer change information | When using the services, if the customer changes their location, update the customer data in the old and new branch, update the region section for the customer. | Easier to serve customers, customers can get to the branch that closest to them, instead of getting to another city. |
| 3 | A customer stops using services. | When a customer stops using a service, change the status in the contract. | Manage customers and services. |
| 4 | Customer pay in account | When a customer pays in, update the balance data in the table SDKH and log the activities in the HDKH table | Let customers pay in their balance. |
| 5 | Customer uses services | When customer use the services, the customer’s balance would be reduced the amount of money that prescribed in the service data | Update activities of customers, when customers have any questions about their activities, they can easily query. |
| 6 | Move customer to the blacklist | Enter customer name and code, update the blacklisted view | Easily manage blacklisted customers. |
| 7 | New customer | When there is a new customer, new data would be inserted, the new  customer’s default balance would be  50000 VND |  |