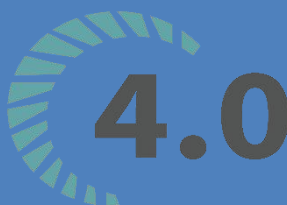


BỘ MÔN HỆ THỐNG THÔNG TIN – KHOA CÔNG NGHỆ THÔNG TIN  
ĐẠI HỌC KHOA HỌC TỰ NHIÊN TP.HCM, ĐẠI HỌC QUỐC GIA TP.HCM

# MÔN CƠ SỞ DỮ LIỆU NÂNG CAO



Sinh viên thực hiện:

19126053-Dương Quang Minh

19126044-Lê Trung Hiếu

19126054-Nguyễn Gia Minh

18126020- Trần Bảo Khánh

GV phụ trách: TS THÁI LÊ VINH

HỒ THỊ HOÀNG VY

ĐỒ ÁN/BÀI TẬP MÔN HỌC - CƠ SỞ DỮ LIỆU NÂNG CAO  
HỌC KỲ III – NĂM HỌC 2021-2022



## BẢNG THÔNG TIN CHI TIẾT NHÓM

<b>Mã nhóm:</b>	<b>06</b>			
<b>Tên nhóm:</b>				
<b>Số lượng:</b>	<b>4</b>			
<b>MSSV</b>	<b>Họ tên</b>	<b>Email</b>	<b>Điện thoại</b>	<b>Hình ảnh</b>
18126044	Lê Trung Hiếu	19126044@student.hcmus.edu.vn		
18126053	Dương Quang Minh	19126053@student.hcmus.edu.vn		
18126054	Nguyễn Gia Minh	19126054@student.hcmus.edu.vn		
18126020	Trần Bảo Khánh	18126006@student.hcmus.edu.vn		

<b>Bảng phân công &amp; đánh giá hoàn thành công việc</b>			
<b>Công việc thực hiện</b>	<b>Người thực hiện</b>	<b>Mức độ hoàn thành</b>	<b>Đánh giá của nhóm</b>
Đặc tả đồ án	Cả nhóm	100%	10/10
Quy trình nghiệp vụ	Cả nhóm	100%	10/10
Toàn vẹn dữ liệu	Trần Bảo Khánh	100%	10/10
Yêu cầu chức năng	Cả nhóm	100%	10/10
Xác định yêu cầu dữ liệu	Cả nhóm	100%	10/10
Layout	Lê Trung Hiếu Nguyễn Gia Minh	100%	10/10
Xác định chỉ mục	Dương Quang Minh Trần Bảo Khánh	100%	10/10
Cài đặt chỉ mục	Dương Quang Minh	100%	10/10
Phân tích	Dương Quang Minh Nguyễn Gia Minh	100%	10/10
Kiểm chứng hiệu quả của chỉ mục	Nguyễn Gia Minh Lê Trung Hiếu	100%	10/10
Partition	Dương Quang Minh Lê Trung Hiếu	100%	10/10
<b>Chức năng</b>			
Trang admin <ul style="list-style-type: none"><li>- Thống kê thu - chi</li><li>- Xem danh sách user trong hệ thống, tìm kiếm theo tên</li><li>- Xem danh sách, thông tin, tìm kiếm các nhân</li></ul>	Nguyễn Gia Minh Lê Trung Hiếu	100%	10/10



viên.			
Chức năng của nhân viên: <ul style="list-style-type: none"><li>- Xem thống kê thu - chi</li><li>- Xem, chỉnh sửa thông tin cá nhân</li><li>- Xem thông tin của khách hàng và chỉnh sửa</li><li>- Thêm thông tin, tạo tour mới vào hệ thống</li><li>- Quản lý thông tin trạng thái tour</li></ul>	Dương Quang Minh Trần Bảo Khánh Nguyễn Gia Minh Lê Trung Hiếu	100%	10/10
Trang chủ: <ul style="list-style-type: none"><li>- Xem danh sách tour trên trang chủ</li><li>- Hiện thị theo danh sách, tìm kiếm theo tên tour</li><li>- Đăng nhập, đăng ký</li></ul>	Nguyễn Gia Minh Lê Trung Hiếu	100%	10/10
User: <ul style="list-style-type: none"><li>- Xem thông tin cá nhân, thay đổi thông tin cá nhân</li><li>- Xem thông tin các tour, xem danh sách các tour đã đặt, lịch sử đặt tour</li></ul>	Trần Bảo Khánh Dương Quang Minh Lê Trung Hiếu Nguyễn Gia Minh	100%	10/10



## YÊU CẦU ĐỒ ÁN- BÀI TẬP

Loại bài tập	<input checked="" type="checkbox"/> Lý thuyết    Thực hành    Đồ án <input checked="" type="checkbox"/> Bài tập
Ngày bắt đầu	
Ngày kết thúc	

### Contents

Yêu cầu của Đồ án/Bài tập.....**Error! Bookmark not defined.**

<b>1. Mô Tả:</b> .....	5
a.Quy trình nghiệp vụ.....	5
1)Quy trình cho khách hàng:.....	5
2)Quy trình cho nhân viên (Bao gồm cả admin lẫn nhân viên thông thường):.....	5
3)Quy trình thanh toán:.....	5
b.Danh sách chức năng	
<b>2. Thiết kế dữ liệu quan niệm</b> .....	6
a. Đặc tả hệ thống.....	6
b. Ràng buộc dữ liệu.....	7
<b>3. Thiết kế dữ liệu logic</b> .....	10
a. Mô hình ER.....	10
b. Mô hình quan hệ.....	10
c. Chuẩn hóa dữ liệu:.....	11
<b>4. Thiết kế vật lý (thông tin tần suất, cài chỉ mục, phân tích, kiểm chứng hiệu quả chỉ mục...)</b>	15
a. Thông tin tần suất.....	17
b. Chỉ mục.....	17
c. Phân tích.....	18
d. Kiểm chứng hiệu quả chỉ mục.....	19
e. Trigger và Procedure.....	26
<b>5. Xác định yêu cầu chức năng:</b> .....	36
<b>6. Layout của giao diện:</b> .....	39
<b>7. Partiton</b> .....	41



<b>I.</b>	<b>Partition:</b> .....	48
<b>II.</b>	<b>Lợi ích của việc sử dụng Partition:</b> .....	48
<b>III.</b>	<b>Các kiểu phân chia dữ liệu:</b> .....	48
<b>IV.</b>	<b>Phân chia dữ liệu bằng Order (Hoá đơn) theo chiều ngang:</b> .....	49

## Mô tả

### A/Quy trình nghiệp vụ

#### 1) Quy trình cho người mua:

- Đăng nhập tài khoản.
- Tìm kiếm, tham khảo thông tin tour, dịch vụ, khuyến mại.
- Người Mua đưa ra quyết định đặt tour bằng cách click vào “Đặt hàng”.
- Người Mua lựa chọn cách thức giao dịch tùy theo chính sách bên bán.
- Đơn hàng của Người Mua sẽ được chuyển thông tin đến bên bán.
- Người Mua nhận được thông tin xác nhận về đơn hàng
- Người Mua chỉnh sửa đơn hàng.

#### 2) Quy trình cho bên cung cấp tour:

- Đăng nhập tài khoản
- Tạo tài khoản (chỉ dành cho nhân viên có quyền admin)
- Tìm kiếm, xem thông tin, cập nhật, tạo các tour
- Quản lý đơn hàng
- Sửa bảng trạng thái của tour
- Thống kê thông tin
- Gửi thông báo đến người mua

#### 3) Quy trình thanh toán:

- + Khách hàng đặt tour -> chuyển thông tin đơn hàng đến nhân viên chăm sóc khách hàng.
- + Nhân viên liên hệ khách hàng để xác nhận thông tin tour, thông tin khách hàng, phương thức thanh toán cho khách hàng.
- + Xác nhận đơn hàng, chuyển trạng thái của đơn hàng sang “Chờ thanh toán”
- + Xác nhận thông tin, hóa đơn thanh toán, chuyển trạng thái của đơn hàng sang “Đã thanh toán”.

### B/Danh sách yêu cầu chức năng

- Đăng nhập/Đăng ký user
- Tìm lại password
- Hiển thị sản phẩm
- Thêm sản phẩm vào giỏ hàng
- Thêm/xóa/ cập nhật thông tin sản phẩm
- Order sản phẩm
- Chỉnh sửa thông tin user
- Sắp xếp hiển thị sản phẩm theo loại sản phẩm

#### 1. Yêu cầu phi chức năng

- Toàn vẹn dữ liệu
- Khả năng tương tác
- Khả năng sử dụng

- Khả năng quản lí
- Khả năng bảo trì

## Thiết kế dữ liệu quan niệm

### A/.Đặc tả hệ thống

Công ty MHMK mở một trang buôn bán, quản lý các tour du lịch. Công ty cần xây dựng 1 trang web cho phép công ty và các cá nhân thực hiện việc mua bán các tour. Sau đây là kết quả của việc phân tích yêu cầu ứng dụng:

- Trang MHMKTour bán đủ loại tour với các tour do công ty tự bán và quản lý.
- Một **nhân viên** có: *mã nhân viên, tên tài khoản, tên nhân viên*, với nhân viên không phải là admin, thì có thêm thông tin về Admin quản lý nhân viên đó. Với nhân viên có cấp bậc là admin có quyền tạo các tài khoản nhân viên và quy định cấp bậc cho họ. Với các nhân viên còn lại có quyền thay đổi, cập nhật, tạo các tour.
- Cần lưu trữ **thông tin khách hàng**, gồm : *mã khách hàng, tên khách hàng, số điện thoại, địa chỉ, email* . Mỗi khách hàng chỉ được sử dụng 1 số điện thoại duy nhất để đăng ký.
- Cần lưu lại thông tin về tất cả các tour mà công ty MHMK cung cấp gồm: *Mã tour, ngày bắt đầu, ngày kết thúc, mô tả, nơi bắt đầu, nơi tham quan, phương tiện di chuyển, giá thành*. Mỗi tour có một nơi bắt đầu tour gồm : *mã địa điểm, tên nơi bắt đầu*. Ngoài ra, các tour còn có nơi tham quan gồm: *mã địa điểm, tên nơi tham quan* .
- Khi khách hàng chọn tour, thì sản phẩm sẽ được đưa vào **giỏ hàng** của khách hàng đó, có: Mã đơn, Mã khách hàng và Tổng tiền, bên cạnh đó, các thông tin chi tiết của đơn hàng sẽ được lưu riêng biệt bao gồm: Mã tour, số lượng người lớn, số lượng trẻ em. Ngay sau khi khách hàng đặt mua thì trạng thái của đơn hàng sẽ được chuyển thành *chưa thanh toán*. Sau khi nhận được thông tin đơn hàng, nhân viên sẽ căn cứ vào số điện thoại của khách hàng được lưu trên thông tin thanh toán sẽ gọi để trao đổi thêm thông tin với khách hàng và thông tin tới khách hàng về việc thanh toán. Khi xác nhận thông tin thanh toán của khách hàng, nhân viên sẽ chuyển trạng thái của đơn hàng thành *đã thanh toán*, Tuy nhiên, nếu như khách hàng hủy hoặc quá thời hạn giữ đơn hàng mà khách hàng chưa thanh toán, đơn sẽ được hủy, nhưng để đảm bảo việc phục hồi đơn khi cần thiết, đơn hàng sẽ không được xóa hoàn toàn mà sẽ được đưa ra khỏi các tiến trình xử lý.
- MHMKTour cần lưu lại các thông tin giao dịch vào **nhật kí giao dịch** và lưu trữ theo: *mã giao dịch, mã đơn hàng, ngày lập giao dịch*. Tuy nhiên, chỉ các đơn hàng có trạng thái là *đã thanh toán* thì mới được đưa vào nhật ký giao dịch.

## B/.Ràng buộc dữ liệu

- Admin:**

<b>Mô tả</b>		
<b>Dữ liệu</b>	<b>Kiểu dữ liệu</b>	<b>Ràng buộc</b>
IdAdmin	int	PRIMARY Key
Username	nvarchar(50)	NOT NULL
Password	nvarchar(50)	NOT NULL

- Staff:**

<b>Mô tả</b>		
<b>Dữ liệu</b>	<b>Kiểu dữ liệu</b>	<b>Ràng buộc</b>
IdStaff	int	PRIMARY Key
IdAdmin	int	FOREIGN KEY
Fullname	nvarchar(50)	NOT NULL
Username	nvarchar(50)	NOT NULL
Password	nvarchar(50)	NOT NULL

- Client:**

<b>Mô tả</b>		
<b>Dữ liệu</b>	<b>Kiểu dữ liệu</b>	<b>Ràng buộc</b>
IdClient	int	PRIMARY Key
Username	nvarchar(32)	NOT NULL
Password	nvarchar(32)	NOT NULL
Fullname	nvarchar(10)	NOT NULL
Phone	nvarchar(10)	UNIQUE
Address	nvarchar(50)	
Email	nvarchar(50)	

- Tour:**

<b>Mô tả</b>		
<b>Dữ liệu</b>	<b>Kiểu dữ liệu</b>	<b>Ràng buộc</b>
IdTour	int	PRIMARY Key
TourName	nvarchar(50)	UNIQUE
DayStart	Date	NOT NULL



DayEnd	Date	NOT NULL
Description	nvarchar(10)	NOT NULL
LocationStart	nvarchar(10)	FOREIGN KEY
Vehicle	nvarchar(50)	NOT NULL
Cost	float	NOT NULL

- Description:**

<u>Mô tả</u>		
Dữ liệu	Kiểu dữ liệu	Ràng buộc
IdDescription	int	PRIMARY Key
Tour	int	FOREIGN Key
Total	float	NOT NULL
createDate	Datetime	NOTNULL

- DescriptionDetail:**

<u>Mô tả</u>		
Dữ liệu	Kiểu dữ liệu	Ràng buộc
IdDescriptionDetail	int	PRIMARY Key
IdDescription	int	FOREIGN Key
nameDescription	nvarchar(50)	NOT NULL
Adults	int	NOT NULL
Childs	int	NOT NULL
Incurred	float	
Total	float	NOT NULL

- OrderTour:**

<u>Mô tả</u>		
Dữ liệu	Kiểu dữ liệu	Ràng buộc
IdOrder	int	PRIMARY Key
IdClient	int	FOREIGN Key
Total	float	NOT NULL

- OrderDetail:**

<u>Mô tả</u>		
Dữ liệu	Kiểu dữ liệu	Ràng buộc

id	int	PRIMARY KEY
IdOrder	int	FOREIGN KEY
IdTour		FOREIGN KEY
Adults	int	NOT NULL
Childrens	int	NOT NULL

- **OrderStatus:**

Mô tả		
Dữ liệu	Kiểu dữ liệu	Ràng buộc
Id	int	PRIMARY KEY
IdOrder	int	FOREIGN KEY
Status	char	NOT NULL

- **Trash:**

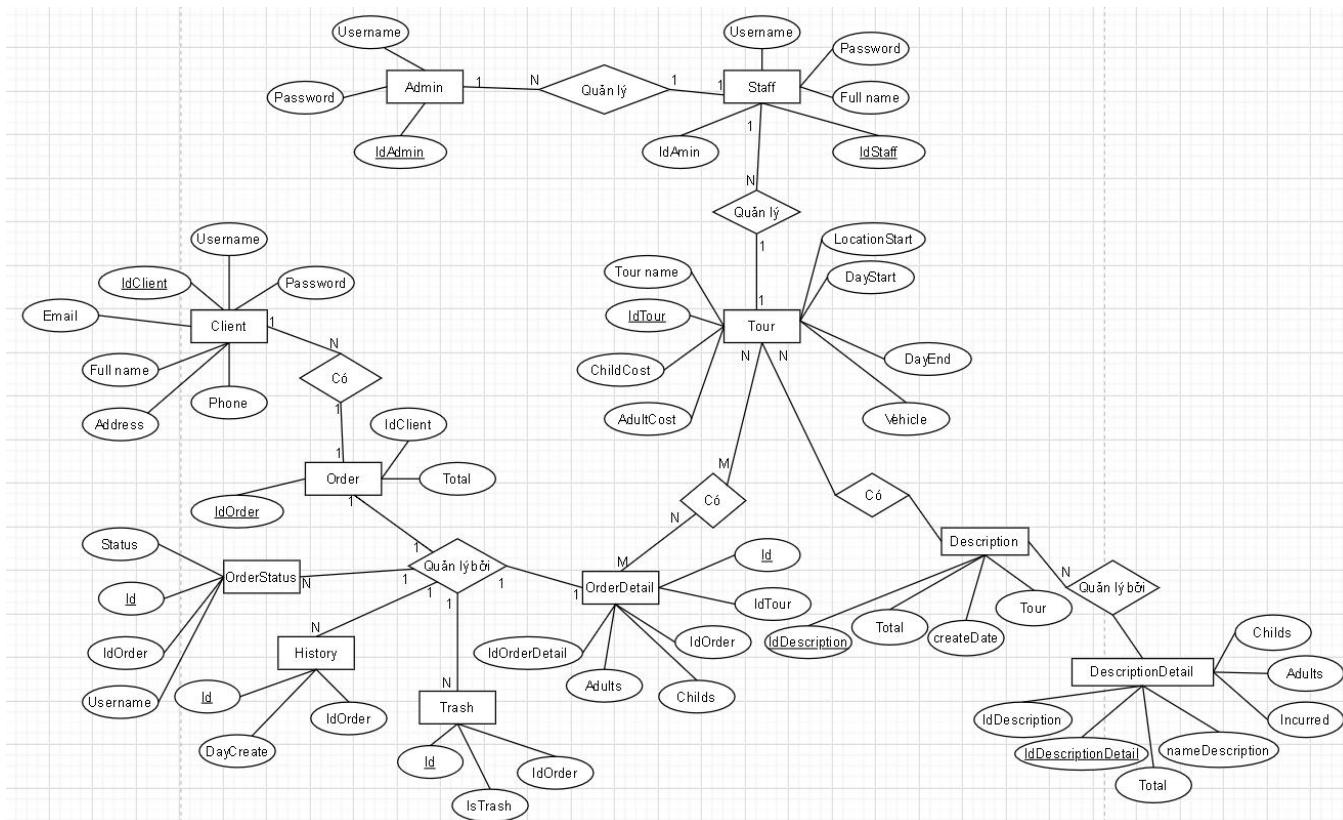
Mô tả		
Dữ liệu	Kiểu dữ liệu	Ràng buộc
Id	int	PRIMARY KEY
IdOrder	int	FOREIGN KEY
IsTrash	boolean	NOT NULL

- **History:**

Mô tả		
Dữ liệu	Kiểu dữ liệu	Ràng buộc
Id	int	PRIMARY KEY
IdOrder	int	FOREIGN KEY
CreateDate	DateTime	NOT NULL

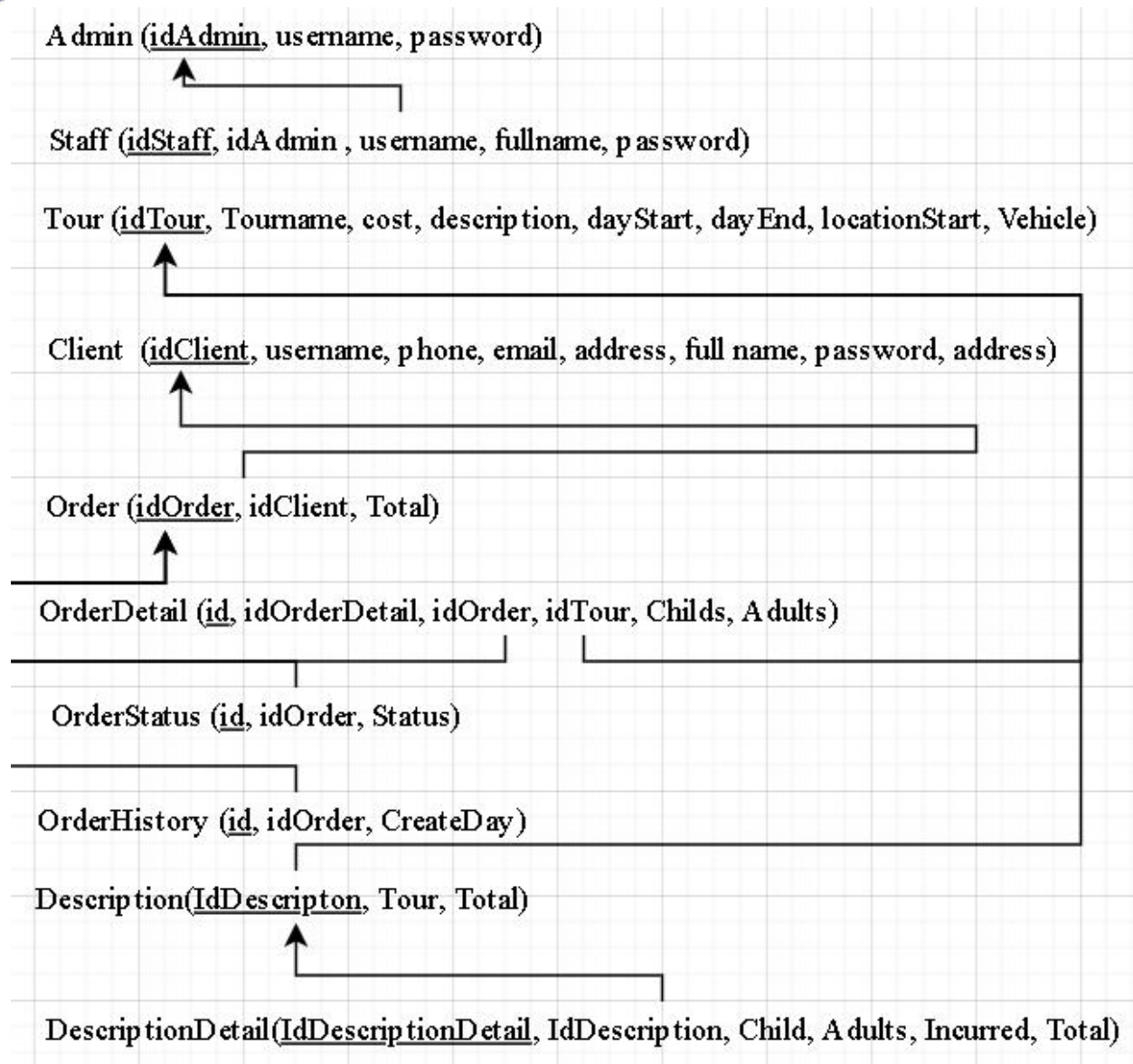
# Thiết kế dữ liệu logic

## 1. Mô hình ER



Mô hình ER được xây dựng dựa trên cách thức vết dầu loang, với trung tâm là Tour.

## 2. Mô hình quan hệ



### 3.Chuẩn hóa dữ liệu:

- Client

Client (idClient, username, phone, email, address, full name, password) không thuộc dạng chuẩn.

Trong bảng này có một trường là thuộc tính đa trị : address

⇒ Bảng mới sau khi nâng chuẩn sẽ là

Client (idClient, username, phone, email, full name, password)

Address (id, idClient, number, street, wards, district, city )

⇒ Bảng Customer sẽ ở dạng chuẩn 3NF :

- Là 2NF
- Không có phụ thuộc hàm bắc cầu

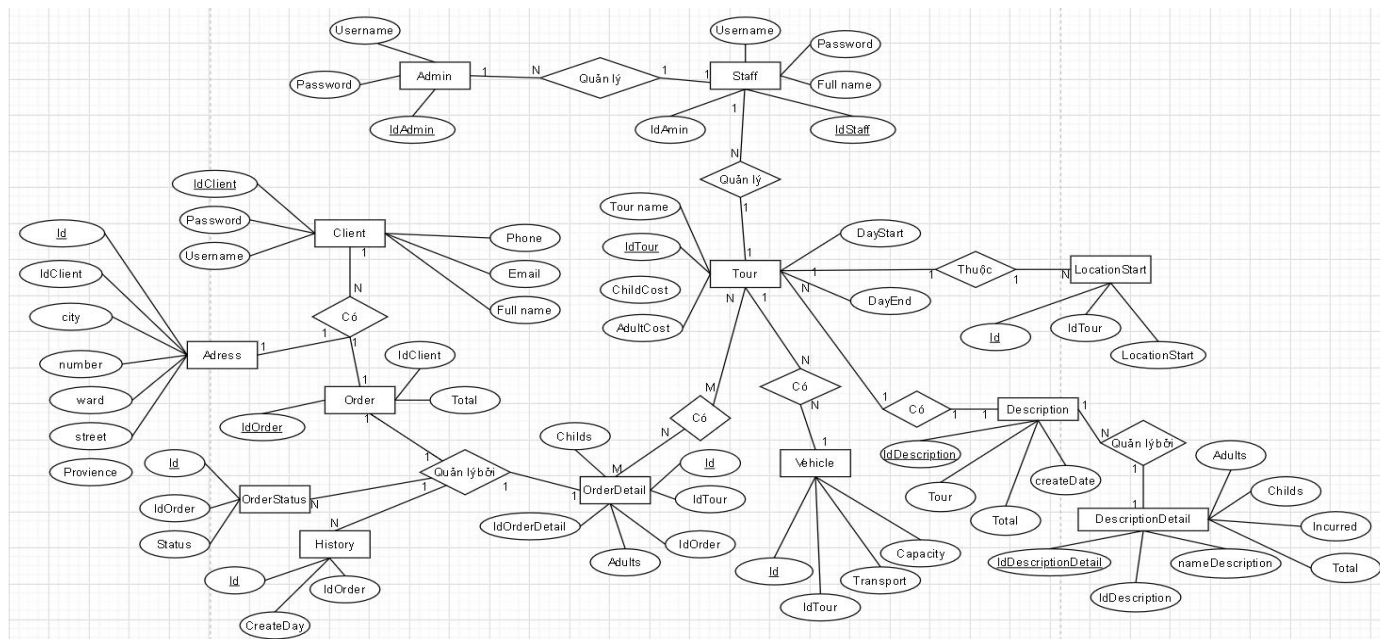
- Tour  
Tour (idTour, Tourname, cost, description, vehicle, dayStart, dayEnd, LocationStart)  
Trong bảng này có hai trường là thuộc tính đa trị : Vehicle, LocationStart  
⇒ Bảng mới sau khi nâng chuẩn sẽ là :  
Tour (idTour, Tourname, cost, description, IdVehicle, dayStart, dayEnd)  
LocationStart(id, idTour, Location)  
Vehicle(id, idTour, Vehicle, Capacity)  
⇒ Bảng Customer sẽ ở dạng chuẩn 3NF :
  - Là 2NF
  - Không có phụ thuộc hàm bắc cầu
- Admin  
Admin (idAdmin, username, password)  
⇒ Bảng Admin sẽ ở dạng chuẩn 3NF:
  - Là 2NF
  - Không có phụ thuộc hàm bắc cầu
- Staff  
Staff (idStaff, username, fullname, password)  
⇒ Bảng Staff sẽ ở dạng chuẩn 3NF:
  - Là 2NF
  - Không có phụ thuộc hàm bắc cầu
- Order  
Order (idOrder, idClient, Total)  
⇒ Bảng Order sẽ ở dạng chuẩn 3NF:
  - Là 2NF
  - Không có phụ thuộc hàm bắc cầu
- OrderDetail  
OrderDetail (id, idOrderDetail, idOrder, idTour, Childs, Adults)  
⇒ Bảng OrderDetail sẽ ở dạng chuẩn 3NF:
  - Là 2NF
  - Không có phụ thuộc hàm bắc cầu
- OrderStatus  
OrderStatus (id, idOrder, Status)  
⇒ Bảng OrderStatus sẽ ở dạng chuẩn 3NF:
  - Là 2NF
  - Không có phụ thuộc hàm bắc cầu
- OrderHistory  
OrderHistory (id, idOrder, CreateDay)

⇒ Bảng Admin sẽ ở dạng chuẩn 3NF:

- Là 2NF
- Không có phụ thuộc hàm bắc cầu

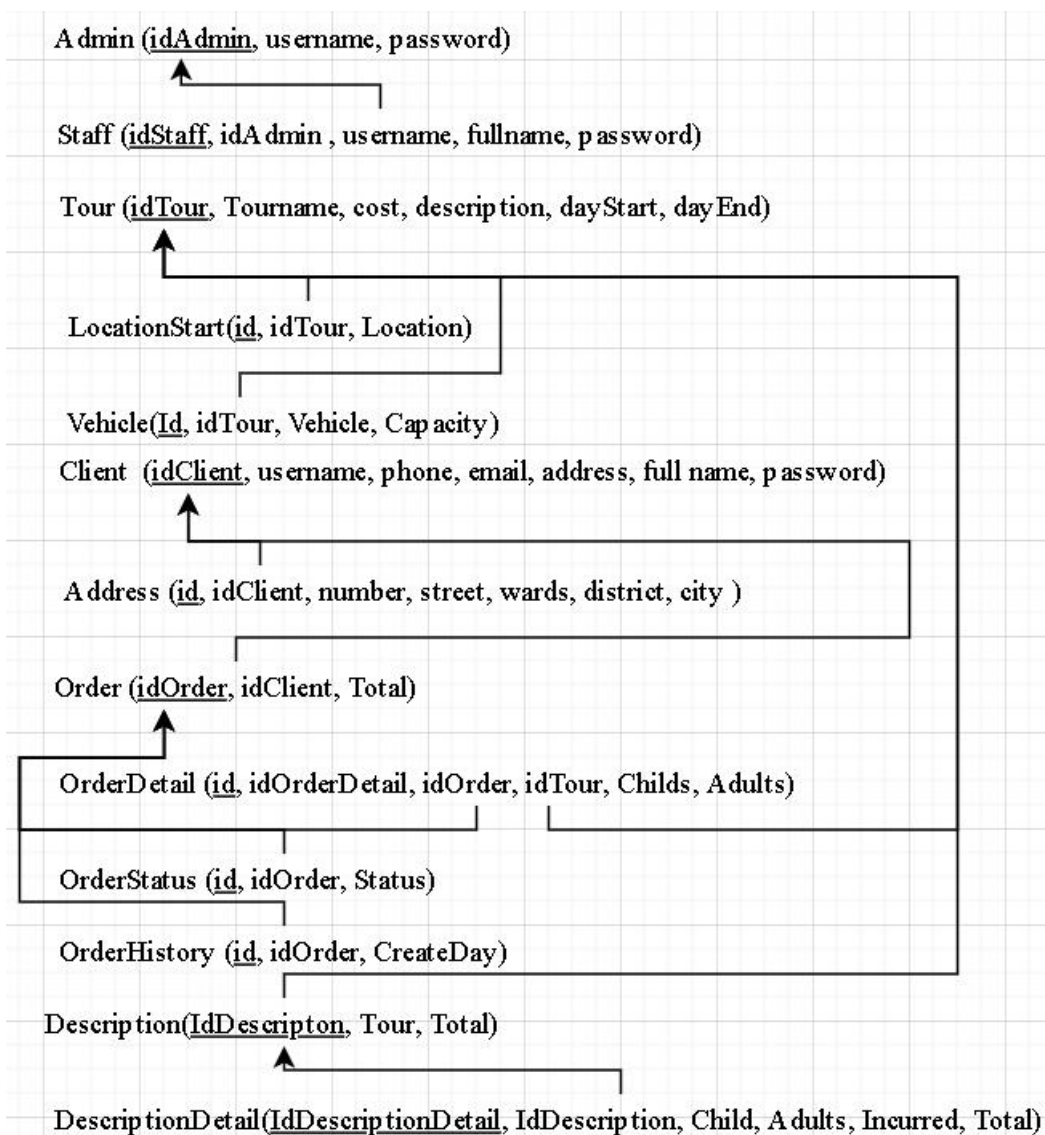
4. ERD, lược đồ quan hệ và miêu tả các trường dữ liệu sau khi được chuẩn hóa.

a. Mô hình ERD



b. Lược đồ quan hệ





c. Mô tả các trường dữ liệu bổ sung

- Tour:

Mô tả		
Dữ liệu	Kiểu dữ liệu	Ràng buộc
IdTour	int	PRIMARY Key
TourName	nvarchar(50)	UNIQUE
DayStart	Date	NOT NULL
DayEnd	Date	NOT NULL

Description	nvarchar(10)	NOT NULL
Cost	float	NOT NULL

- LocationStart:**

Mô tả		
Dữ liệu	Kiểu dữ liệu	Ràng buộc
IdLocation	int	PRIMARY Key
Province	nvarchar	NOT NULL
Tour	char	FOREIGNKEY

- Vehicle:**

Mô tả		
Dữ liệu	Kiểu dữ liệu	Ràng buộc
Id	char	PRIMARY Key
Vehicle	char	NOT NULL
Capacity	int	NOT NULL
Tour	char	FOREIGNKEY

- Client:**

Mô tả		
Dữ liệu	Kiểu dữ liệu	Ràng buộc
IdClient	int	PRIMARY Key
Username	nvarchar(32)	UNIQUE
Password	nvarchar(32)	NOT NULL
Fullname	nvarchar(10)	NOT NULL
Phone	nvarchar(10)	UNIQUE
Email	nvarchar(50)	

- Address**

Mô tả		
Dữ liệu	Kiểu dữ liệu	Ràng buộc
id	int	PRIMARY Key
username	int	FOREIGNKEY
number	int	NOT NULL





street	Nvarchar(32)	NOTNULL
ward	Nvarchar(32)	NOTNULL
city	Nvarchar(32)	NOTNULL
provience	Nvarchar(32)	NOTNULL

# Thiết kế vật lý (thông tin tần suất, cài chỉ mục, phân tích, kiểm chứng hiệu quả chỉ mục...)

## 1. Thông tin tần suất

Chức năng	Mục tiêu	Tần suất (lượt truy vấn)
Tìm kiếm thông tin khách hàng	Tìm theo tên hoặc số điện thoại của khách hàng	1M+/- ngày
Tìm kiếm thông tin nhân viên (Staff)	Tìm theo tên hoặc tên tài khoản	10K/ngày
Tìm kiếm thông tin nhân viên (Admin)	Tìm theo tên tài khoản	10K/ngày
Tìm kiếm thông tin tour	Tìm theo tên tour hoặc tìm theo những nơi tham quan	10M+/- ngày
Hiển thị thông tin giỏ hàng	Hiển thị giỏ hàng theo mã khách hàng	500k/ ngày
Hiển thị đơn hàng	Hiển thị theo mã khách hàng hoặc ngày tạo order	500k/ngày
Hiển thị lịch sử giao dịch của KH	Tìm theo mã khách hàng và ngày tạo log	500K/ ngày
Hiển thị trạng thái của các đơn hàng	Tìm theo tình trạng đơn hàng	500K/ ngày
Hiển thị doanh thu mà ncc thu được	Tìm kiếm theo ngày tạo và mã Order	500K/ ngày
Hiển thị mức chi mà ncc phải chi	Tìm kiếm theo mã Order, ngày tạo, thông tin, trạng thái tour	500K/ ngày

### a. Chỉ mục

-- Index cho Admin

CREATE INDEX TimKiemAdmin ON Admin (Username ASC)

-- Index cho nhân viên

CREATE NONCLUSTERED INDEX TimKiemNhanVien ON Staff (

Fullname ASC,

Username ASC)

--Index cho khách hàng

CREATE NONCLUSTERED INDEX TimKiemKhachHang ON Client



(Fullname ASC,

Username ASC,

Phone ASC,

Email ASC)

-- Index cho danh sách Tour

CREATE NONCLUSTERED INDEX TimKiemTour ON Tour (TourName ASC,  
DayStart ASC, DayEnd ASC)

-- Index cho nơi bắt đầu

CREATE INDEX TimKiemLocationStart\_byLocation ON LocationStart (Province  
ASC)

CREATE INDEX TimKiemLocationStart\_byTour ON LocationStart (Tour ASC)

-- Index cho nơi tham quan

CREATE INDEX TimKiemLocationVisit\_byLocation ON LocationVisit (VisitPlace  
ASC)

CREATE INDEX TimKiemLocationVisit\_byTour ON LocationVisit (Tour ASC)

-- Index cho phương tiện

CREATE INDEX TimKiemPhuongTien ON Vehicle (Transport)

-- Index cho đơn hàng

CREATE NONCLUSTERED INDEX TimKiemOrder ON OrderTour(IdClient ASC,  
createDate ASC)

-- Index cho trạng thái đơn hàng

CREATE NONCLUSTERED INDEX TimKiemOrderStatus ON OrderStatus  
(IdOrder ASC, OrderStatus ASC)

## b. Phân tích

- Tìm kiếm thông tin nhân viên theo tên hoặc username: do tần suất tìm kiếm nhân viên để thực hiện công việc quản lý, nên ta cài index để cải thiện performance cho hệ thống
- Tìm kiếm thông tin khách hàng theo tên, username hoặc số điện thoại: do tần suất tìm kiếm khách hàng để tạo order hoặc xem thông tin là cao, nên ta cài index để cải thiện performance cho hệ thống.

- Tìm kiếm tour là chức năng có lượt truy vấn nhiều nhất trên hệ thống, vì vậy phải cải thiện performance cho việc truy vấn liên quan đến sản phẩm
- Bên cạnh việc tìm kiếm tour dựa trên TourName, người dùng còn có thể tìm kiếm dựa trên phương tiện, nơi bắt đầu, nơi tham quan vì thế ta cài index ở những bảng Vehicle, LocationStart, LocationVisit để cải thiện performance cho việc truy vấn liên quan đến thông tin tour
- Hiện thị đơn hàng: hiển thị theo id khách hàng hoặc ngày tạo order dùng để kiểm soát thông tin order của khách trong việc thanh toán hoặc kiểm tra doanh thu, lịch sử giao dịch của khách hàng.
- Các bảng kiểm tra lịch sử giao dịch được admin và nhà cung cấp xem lại hàng ngày/tháng/năm để kiểm soát doanh thu và cải thiện việc bán hàng. Vì vậy cần cài đặt index ở các bảng này để việc truy suất dữ liệu được nhanh hơn (do dữ liệu lưu trữ rất lớn).

### c. Kiểm chứng hiệu quả chỉ mục

Kích thước dữ liệu dùng để kiểm chứng hiệu quả của chỉ mục

1. Admin : 1.000 dòng dữ liệu
2. Nhân viên (Staff): 150.000 nhân viên
3. Khách hàng : 500.000 khách hàng
4. Tour : 1.000.000 sản phẩm
5. Vehicle: 1.500.000 phương tiện
6. LocationStart: 1.000.000 vị trí
7. LocationVisit: 4.000.000 vị trí
8. Order: 800.000 đơn hàng
9. OrderStatus: 800.000 đơn hàng

Kết quả đánh giá một số Index (Index cho bảng Staff và bảng Client)

- Staff

Kiểm chứng hiệu quả việc sử dụng index dựa trên kết quả tìm kiếm nhân viên theo quản trị viên quản lý:

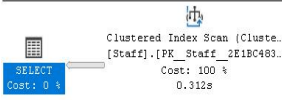
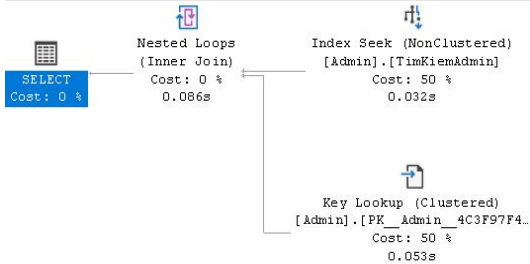
Query: **Select \* from Staff where Admin = 'Admin667'**

Kết quả:

	IdStaff	Admin	FullName	Username	Password	Status
1	28341	Admin667	Lã Xuân Mai	Staff28340	pass28340	1
2	42142	Admin667	Lêng Phúc Nguyên	Staff42141	pass42141	1
3	60300	Admin667	Cái Hải My	Staff60299	pass60299	1
4	67241	Admin667	Đoạn Bích Hà	Staff67240	pass67240	1
5	85806	Admin667	Phi Xuân Hiền	Staff85805	pass85805	1
6	87646	Admin667	Mạch Kim Cường	Staff87645	pass87645	1
7	93055	Admin667	Công Hoàng Sa	Staff93054	pass93054	1
8	93983	Admin667	Tôn Thất Tú Quyền	Staff93982	pass93982	1
9	112202	Admin667	Khoa Hải San	Staff112202	pass112202	0
10	119162	Admin667	Hàn Đan Thu	Staff119162	pass119162	0
11	131942	Admin667	Thẩm Như Trần	Staff131942	pass131942	0
12	144002	Admin667	Vũ Thực Trinh	Staff144002	pass144002	0
13	150022	Admin667	Tôn Thất Hạnh San	Staff150022	pass150022	0
14	157640	Admin667	Dã Thế Minh	Staff157640	pass157640	0
15	163446	Admin667	Cà Hạnh San	Staff163446	pass163446	0
16	181183	Admin667	Bình Bảo Hoa	Staff181183	pass181183	0
17	184478	Admin667	Bá Minh Huyền	Staff184478	pass184478	0
18	192330	Admin667	Ấu Trung Chuyên	Staff192330	pass192330	0



Excution plan:

Excution plan không dùng index	Excution plan dùng index
<p>Query 1: Query cost (relative to the batch): 100%</p> <p>SELECT * FROM [dbo].[Staff] WHERE [Admin]=@1</p> <p>Missing Index (Impact 99.7823): CREATE NONCLUSTERED INDEX [&lt;Name of Missing Index, sysname,&gt;] ON [dbo].[Staff] ([Admin])</p> 	<p>Query 1: Query cost (relative to the batch): 100%</p> <p>SELECT * FROM [dbo].[Admin] WHERE [Username]=@1</p> 

Hiệu suất:

Clustered Index Scan (Clustered)	Index Seek (NonClustered)
Scanning a clustered index, entirely or only a range.	Scan a particular range of rows from a nonclustered index.
<b>Physical Operation</b> Clustered Index Scan	<b>Physical Operation</b> Index Seek
<b>Logical Operation</b> Clustered Index Scan	<b>Logical Operation</b> Index Seek
<b>Actual Execution Mode</b> Row	<b>Actual Execution Mode</b> Row
<b>Estimated Execution Mode</b> Row	<b>Estimated Execution Mode</b> Row
<b>Storage</b> RowStore	<b>Storage</b> RowStore
<b>Actual Number of Rows for All Executions</b> 1	<b>Actual Number of Rows for All Executions</b> 1
<b>Number of Rows Read</b> 200000	<b>Estimated Operator Cost</b> 0,0032845 (37%)
<b>Actual Number of Batches</b> 0	<b>Estimated I/O Cost</b> 0,003125
<b>Estimated Operator Cost</b> 1,67958 (100%)	<b>Estimated Subtree Cost</b> 0,0032845
<b>Estimated I/O Cost</b> 1,45942	<b>Estimated CPU Cost</b> 0,0001595
<b>Estimated Subtree Cost</b> 1,67958	<b>Number of Executions</b> 1
<b>Estimated CPU Cost</b> 0,220157	<b>Estimated Number of Executions</b> 1
<b>Estimated Number of Executions</b> 1	<b>Estimated Number of Rows Per Execution</b> 2,28167
<b>Number of Executions</b> 1	<b>Estimated Number of Rows for All Executions</b> 2,28167
<b>Estimated Number of Rows to be Read</b> 200000	<b>Estimated Number of Rows to be Read</b> 2,28167
<b>Estimated Number of Rows for All Executions</b> 1,85074	<b>Estimated Row Size</b> 103 B
<b>Estimated Number of Rows Per Execution</b> 1,85074	<b>Ordered</b> True
<b>Estimated Row Size</b> 130 B	<b>Node ID</b> 1
<b>Actual Rebinds</b> 0	<b>Object</b> [ManageTourist].[dbo].[Staff].[TimKiemNhanVien]
<b>Actual Rewinds</b> 0	<b>Output List</b> [ManageTourist].[dbo].[Staff].[IdStaff], [ManageTourist].[dbo].[Staff].[Admin], [ManageTourist].[dbo].[Staff].[FullName], [ManageTourist].[dbo].[Staff].[Username]
<b>Ordered</b> False	<b>Seek Predicates</b> Seek Keys[1]: Prefix: [ManageTourist].[dbo].[Staff].[Admin] = Scalar Operator('Admin667')
<b>Node ID</b> 0	
<b>Predicate</b> [ManageTourist].[dbo].[Staff].[Admin]='Admin667'	
<b>Object</b> [ManageTourist].[dbo].[Staff].[PK_Staff_2E1BC483E116B01D]	
<b>Output List</b> [ManageTourist].[dbo].[Staff].[IdStaff], [ManageTourist].[dbo].[Staff].[Admin], [ManageTourist].[dbo].[Staff].[FullName], [ManageTourist].[dbo].[Staff].[Username], [ManageTourist].[dbo].[Staff].[Password]	

Sử dụng câu lệnh select cho bảng Staff ta thấy trước và sau khi sử dụng Index có sự cải thiện rõ rệt về hiệu quả của câu lệnh

Kiểm chứng hiệu quả việc sử dụng index dựa trên kết quả tìm kiếm nhân viên theo tên:

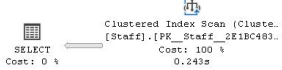
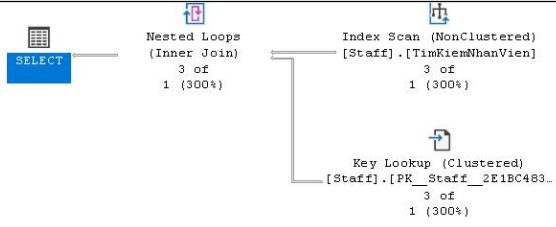

Query: **Select \* from** dbo.Staff **where** FullName = 'Chu Thùy Vân'

Kết quả



	IdStaff	Admin	FullName	Username	Password	Status
1	169	Admin772	Chu Thùy Vân	Staff168	pass168	1
2	37932	Admin4713	Chu Thùy Vân	Staff37931	pass37931	1
3	181794	Admin3813	Chu Thùy Vân	Staff181794	pass181794	0

Excution plan:

Excution plan không dùng index	Excution plan dùng index
<p>Query 1: Query cost (relative to the batch): 100%</p> <p>SELECT * FROM [dbo].[Staff] WHERE ([FullName]=@1</p> <p>Missing Index (Impact 99.7889): CREATE NONCLUSTERED INDEX [&lt;Name of Missing Index, sysname,&gt;] ON [dbo].[Staff] ([FullName])</p> 	 <p>Estimated query: Query 1: Query cost (relative to the batch): 100%</p> <p>SELECT * FROM [dbo].[Staff] WHERE ([FullName]=@1</p> <p>Missing Index (Impact 99.7889): CREATE NONCLUSTERED INDEX [&lt;Name of Missing Index, sysname,&gt;] ON [dbo].[Staff] ([FullName])</p> 

Clustered Index Scan (Clustered)	Key Lookup (Clustered)
Scanning a clustered index, entirely or only a range.	Uses a supplied clustering key to lookup on a table that has a clustered index.
<b>Estimated operator progress: 100%</b>	<b>Estimated operator progress: 100%</b>
<b>Physical Operation</b> Clustered Index Scan	<b>Physical Operation</b> Key Lookup
<b>Logical Operation</b> Clustered Index Scan	<b>Logical Operation</b> Key Lookup
<b>Actual Execution Mode</b> Row	<b>Actual Execution Mode</b> Row
<b>Estimated Execution Mode</b> Row	<b>Estimated Execution Mode</b> Row
<b>Storage</b> RowStore	<b>Storage</b> RowStore
<b>Number of Rows Read</b> 199999	<b>Actual Number of Rows for All Executions</b> 3
<b>Actual Number of Rows for All Executions</b> 3	<b>Estimated Operator Cost</b> 0,0033445 (0%)
<b>Actual Number of Batches</b> 0	<b>Estimated I/O Cost</b> 0,003125
<b>Estimated I/O Cost</b> 1,51498	<b>Estimated Subtree Cost</b> 0,0033445
<b>Estimated Operator Cost</b> 1,73513 (100%)	<b>Estimated CPU Cost</b> 0,0001581
<b>Estimated Subtree Cost</b> 1,73513	<b>Estimated Number of Executions</b> 1,0816311
<b>Estimated CPU Cost</b> 0,220156	<b>Number of Executions</b> 3
<b>Estimated Number of Executions</b> 1	<b>Estimated Number of Rows for All Executions</b> 1,0816311
<b>Number of Executions</b> 1	<b>Estimated Number of Rows Per Execution</b> 1
<b>Estimated Number of Rows for All Executions</b> 1,08163	<b>Estimated Row Size</b> 67 B
<b>Estimated Number of Rows Per Execution</b> 1,08163	<b>Ordered</b> True
<b>Estimated Number of Rows to be Read</b> 199999	<b>Node ID</b> 3
<b>Estimated Row Size</b> 126 B	
<b>Actual Rebinds</b> 0	
<b>Actual Rewinds</b> 0	
<b>Ordered</b> False	
<b>Node ID</b> 0	
<b>Predicate</b> [ManageTourist].[dbo].[Staff].[FullName]='N'Chu Thùy Vân'	<b>Object</b> [ManageTourist].[dbo].[Staff].[PK_Staff_2E1BC483DB231BFC]
<b>Object</b> [ManageTourist].[dbo].[Staff].[PK_Staff_2E1BC483DB231BFC]	<b>Output List</b> [ManageTourist].[dbo].[Staff].Username, [ManageTourist].[dbo].[Staff].Password, [ManageTourist].[dbo].[Staff].Status
<b>Output List</b> [ManageTourist].[dbo].[Staff].IdStaff, [ManageTourist].[dbo].[Staff].Admin, [ManageTourist].[dbo].[Staff].FullName, [ManageTourist].[dbo].[Staff].Username, [ManageTourist].[dbo].[Staff].Password, [ManageTourist].[dbo].[Staff].Status	<b>Seek Predicates</b> Seek Keys[1]: Prefix: [ManageTourist].[dbo].[Staff].IdStaff = Scalar Operator([ManageTourist].[dbo].[Staff].[IdStaff])





- Client

Query: **Select** \* **from** dbo.Client **where** FullName = 'Lò Cát Linh';

Kết quả:

	IdClient	FullName	Phone	Username	Password	Email
1	113281	Lò Cát Linh	1206098488	Client113280	pass113280	client113280@gmail.com
2	187	Lò Cát Linh	5907127663	Client186	pass186	client186@gmail.com

### Excution plan

Excution plan không dùng index	Excution plan với index
<p>Estimated query progress:100% Query 1: Query cost (relative to the batch): 100% Select * from dbo.Client where FullName = 'Lò Cát Linh' Missing Index (Impact 99.885): CREATE NONCLUSTERED INDEX [&lt;Name of Missing Index, sysname,&gt;] ON [dbo].[Client] ([FullName])</p> 	<p>Results Messages Live Query Statistics Execution plan Query 1: Query cost (relative to the batch): 100% SELECT * FROM [dbo].[Client] WHERE [FullName]=@1</p> 



Clustered Index Scan (Clustered)		Index Seek (NonClustered)	
Scanning a clustered index, entirely or only a range.		Scan a particular range of rows from a nonclustered index.	
Estimated operator progress: 100%		Estimated operator progress: 100%	
Physical Operation	Clustered Index Scan	Physical Operation	Index Seek
Logical Operation	Clustered Index Scan	Logical Operation	Index Seek
Estimated Execution Mode	Row	Actual Execution Mode	Row
Storage	RowStore	Storage	RowStore
Actual Number of Rows for All Executions	2	Number of Rows Read	1
Estimated I/O Cost	2,88683	Actual Number of Rows for All Executions	1
Estimated Operator Cost	3,21698 (100%)	Actual Number of Batches	0
Estimated CPU Cost	0,330149	Estimated Operator Cost	0,0032831 (50%)
Estimated Subtree Cost	3,21698	Estimated I/O Cost	0,003125
Number of Executions	1	Estimated Subtree Cost	0,0032831
Estimated Number of Executions	1	Estimated CPU Cost	0,0001581
Estimated Number of Rows for All Executions	1,02242	Number of Executions	1
Estimated Number of Rows Per Execution	1,02242	Estimated Number of Executions	1
Estimated Number of Rows to be Read	299993	Estimated Number of Rows for All Executions	1
Estimated Row Size	149 B	Estimated Number of Rows to be Read	1
Ordered	False	Estimated Number of Rows Per Execution	1
Node ID	0	Estimated Row Size	122 B
<b>Predicate</b>		Actual Rebinds	0
[ManageTourist].[dbo].[Client].[FullName] = N'Lò Cát Linh'		Actual Rewinds	0
<b>Object</b>		Ordered	True
[ManageTourist].[dbo].[Client].[PK_Client_C1961B332C891574]		Node ID	1
<b>Output List</b>		<b>Object</b>	
[ManageTourist].[dbo].[Client].IdClient, [ManageTourist].[dbo].[Client].FullName, [ManageTourist].[dbo].[Client].Phone, [ManageTourist].[dbo].[Client].Username, [ManageTourist].[dbo].[Client].Password, [ManageTourist].[dbo].[Client].Email		[ManageTourist].[dbo].[Client].[TimKiemKhachHang]	
		<b>Output List</b>	
		[ManageTourist].[dbo].[Client].IdClient, [ManageTourist].[dbo].[Client].FullName, [ManageTourist].[dbo].[Client].Phone, [ManageTourist].[dbo].[Client].Username, [ManageTourist].[dbo].[Client].Email	
		<b>Seek Predicates</b>	
		Seek Keys[1]: Prefix: [ManageTourist].[dbo].[Client].FullName = Scalar Operator(N'Hoa An Nhiên')	

Sử dụng câu lệnh select cho bảng Client ta thấy trước và sau khi sử dụng Index có sự cải thiện rõ rệt về hiệu quả của câu lệnh

Query: `Select * from dbo.Client where Email = 'client34698@gmail.com';`

Kết quả:

Results

Messages

Live Query Statistics

Execution plan

Client Statistics

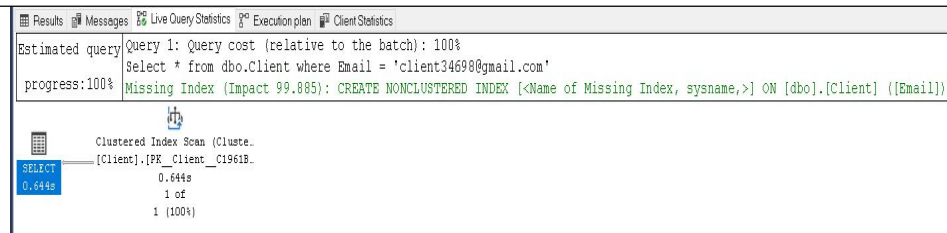
	IdClient	FullName	Phone	Username	Password	Email
1	34699	Thào Hưởng Dương	0764298952	Client34698	pass34698	client34698@gmail.com



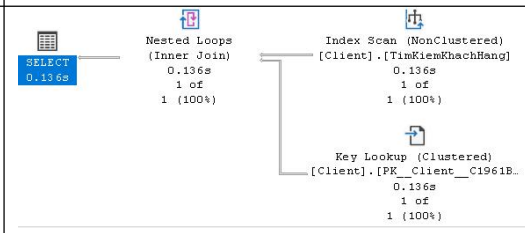


## Excution plan

### Excution plan không dùng index



### Excution plan với index



## Hiệu suất:

Clustered Index Scan (Clustered)	Index Scan (NonClustered)																																																																						
Scanning a clustered index, entirely or only a range.	Scan a nonclustered index, entirely or only a range.																																																																						
Estimated operator progress: 100%	Estimated operator progress: 100%																																																																						
<table><tr><td>Physical Operation</td><td>Clustered Index Scan</td></tr><tr><td>Logical Operation</td><td>Clustered Index Scan</td></tr><tr><td>Estimated Execution Mode</td><td>Row</td></tr><tr><td>Storage</td><td>RowStore</td></tr><tr><td>Actual Number of Rows for All Executions</td><td>1</td></tr><tr><td>Estimated I/O Cost</td><td>2,88683</td></tr><tr><td>Estimated Operator Cost</td><td>3,21698 (100%)</td></tr><tr><td>Estimated CPU Cost</td><td>0,330149</td></tr><tr><td>Estimated Subtree Cost</td><td>3,21698</td></tr><tr><td>Number of Executions</td><td>1</td></tr><tr><td>Estimated Number of Executions</td><td>1</td></tr><tr><td>Estimated Number of Rows for All Executions</td><td>1,01694</td></tr><tr><td>Estimated Number of Rows Per Execution</td><td>1,01694</td></tr><tr><td>Estimated Number of Rows to be Read</td><td>299993</td></tr><tr><td>Estimated Row Size</td><td>169 B</td></tr><tr><td>Ordered</td><td>False</td></tr><tr><td>Node ID</td><td>0</td></tr></table>	Physical Operation	Clustered Index Scan	Logical Operation	Clustered Index Scan	Estimated Execution Mode	Row	Storage	RowStore	Actual Number of Rows for All Executions	1	Estimated I/O Cost	2,88683	Estimated Operator Cost	3,21698 (100%)	Estimated CPU Cost	0,330149	Estimated Subtree Cost	3,21698	Number of Executions	1	Estimated Number of Executions	1	Estimated Number of Rows for All Executions	1,01694	Estimated Number of Rows Per Execution	1,01694	Estimated Number of Rows to be Read	299993	Estimated Row Size	169 B	Ordered	False	Node ID	0	<table><tr><td>Physical Operation</td><td>Index Scan</td></tr><tr><td>Logical Operation</td><td>Index Scan</td></tr><tr><td>Estimated Execution Mode</td><td>Row</td></tr><tr><td>Storage</td><td>RowStore</td></tr><tr><td>Actual Number of Rows for All Executions</td><td>1</td></tr><tr><td>Estimated I/O Cost</td><td>1,77201</td></tr><tr><td>Estimated Operator Cost</td><td>2,10216 (93%)</td></tr><tr><td>Estimated CPU Cost</td><td>0,330149</td></tr><tr><td>Estimated Subtree Cost</td><td>2,10216</td></tr><tr><td>Number of Executions</td><td>1</td></tr><tr><td>Estimated Number of Executions</td><td>1</td></tr><tr><td>Estimated Number of Rows for All Executions</td><td>1,01694</td></tr><tr><td>Estimated Number of Rows Per Execution</td><td>1,01694</td></tr><tr><td>Estimated Number of Rows to be Read</td><td>299993</td></tr><tr><td>Estimated Number of Rows for All Executions</td><td>1,01694</td></tr><tr><td>Estimated Row Size</td><td>88 B</td></tr><tr><td>Ordered</td><td>False</td></tr><tr><td>Node ID</td><td>1</td></tr></table>	Physical Operation	Index Scan	Logical Operation	Index Scan	Estimated Execution Mode	Row	Storage	RowStore	Actual Number of Rows for All Executions	1	Estimated I/O Cost	1,77201	Estimated Operator Cost	2,10216 (93%)	Estimated CPU Cost	0,330149	Estimated Subtree Cost	2,10216	Number of Executions	1	Estimated Number of Executions	1	Estimated Number of Rows for All Executions	1,01694	Estimated Number of Rows Per Execution	1,01694	Estimated Number of Rows to be Read	299993	Estimated Number of Rows for All Executions	1,01694	Estimated Row Size	88 B	Ordered	False	Node ID	1
Physical Operation	Clustered Index Scan																																																																						
Logical Operation	Clustered Index Scan																																																																						
Estimated Execution Mode	Row																																																																						
Storage	RowStore																																																																						
Actual Number of Rows for All Executions	1																																																																						
Estimated I/O Cost	2,88683																																																																						
Estimated Operator Cost	3,21698 (100%)																																																																						
Estimated CPU Cost	0,330149																																																																						
Estimated Subtree Cost	3,21698																																																																						
Number of Executions	1																																																																						
Estimated Number of Executions	1																																																																						
Estimated Number of Rows for All Executions	1,01694																																																																						
Estimated Number of Rows Per Execution	1,01694																																																																						
Estimated Number of Rows to be Read	299993																																																																						
Estimated Row Size	169 B																																																																						
Ordered	False																																																																						
Node ID	0																																																																						
Physical Operation	Index Scan																																																																						
Logical Operation	Index Scan																																																																						
Estimated Execution Mode	Row																																																																						
Storage	RowStore																																																																						
Actual Number of Rows for All Executions	1																																																																						
Estimated I/O Cost	1,77201																																																																						
Estimated Operator Cost	2,10216 (93%)																																																																						
Estimated CPU Cost	0,330149																																																																						
Estimated Subtree Cost	2,10216																																																																						
Number of Executions	1																																																																						
Estimated Number of Executions	1																																																																						
Estimated Number of Rows for All Executions	1,01694																																																																						
Estimated Number of Rows Per Execution	1,01694																																																																						
Estimated Number of Rows to be Read	299993																																																																						
Estimated Number of Rows for All Executions	1,01694																																																																						
Estimated Row Size	88 B																																																																						
Ordered	False																																																																						
Node ID	1																																																																						
<b>Predicate</b> [ManageTourist].[dbo].[Client].[Email]='client34698@gmail.com'	<b>Predicate</b> [ManageTourist].[dbo].[Client].[Email]='client34698@gmail.com'																																																																						
<b>Object</b> [ManageTourist].[dbo].[Client].[PK_Client_C1961B332C891574]	<b>Object</b> [ManageTourist].[dbo].[Client].[TimKiemKhachHang]																																																																						
<b>Output List</b> [ManageTourist].[dbo].[Client].IdClient, [ManageTourist].[dbo].[Client].FullName, [ManageTourist].[dbo].[Client].Phone, [ManageTourist].[dbo].[Client].Username, [ManageTourist].[dbo].[Client].Password, [ManageTourist].[dbo].[Client].Email	<b>Output List</b> [ManageTourist].[dbo].[Client].IdClient, [ManageTourist].[dbo].[Client].FullName, [ManageTourist].[dbo].[Client].Email																																																																						



### ***Đánh giá tổng:***

- Sử dụng index cho kết quả truy vấn tốt hơn, nhanh hơn so với khi không dùng.
- Cải thiện tốc độ khi phải kết nhiều bảng với nhau
- Khi sử dụng index thì các chi phí bỏ ra để truy vấn kết quả đã được giảm đi 1 nửa (sub tree, )
- Những chỉ số khác như I/O, Operator, CPU Cost được giảm đi rõ ràng và có thể thấy được.
- Vậy việc sử dụng index là cần thiết trong việc vận hành hiệu quả ứng dụng, hệ thống, cải thiện tốc độ cũng như hiệu suất truy xuất dữ liệu

#### d. Trigger và Procedure

- Trigger

```
CREATE TRIGGER trg_OrderTour ON OrderDetail AFTER INSERT AS  
  
BEGIN  
  
    UPDATE OrderTour  
  
    SET Total = (SELECT Adluts from OrderDetail where IdOrder = OrderDetail.IdOrder) *  
(SELECT CostAdo from Tour where Tour = Tour.TourName) +  
  
    (Select Childs from OrderDetail where IdOrder = OrderDetail.IdOrder) * (SELECT CostChild  
from Tour where Tour = Tour.TourName)  
  
    FROM OrderTour  
  
    JOIN inserted ON OrderTour.IdOrder = inserted.IdOrder  
  
END  
  
GO
```

```
CREATE TRIGGER trg_CapNhatOrderTour ON OrderDetail After Update as  
  
BEGIN  
  
    UPDATE OrderTour  
  
    SET Total = (SELECT Adluts from OrderDetail where IdOrder = OrderDetail.IdOrder) *  
(SELECT CostAdo from Tour where Tour = Tour.TourName) +  
  
    (Select Childs from OrderDetail where IdOrder = OrderDetail.IdOrder) * (SELECT CostChild  
from Tour where Tour = Tour.TourName)  
  
    FROM OrderTour  
  
    JOIN deleted ON OrderTour.IdOrder = deleted.IdOrder  
  
END  
  
GO
```



```
CREATE TRIGGER trg_HuyOrderTour ON OrderDetail After delete as
```

```
BEGIN
```

```
    UPDATE OrderTour Set Total = 0
```

```
    from OrderTour
```

```
    JOIN deleted ON OrderTour.IdOrder = deleted.IdOrder
```

```
END
```

```
GO
```

```
CREATE TRIGGER trg_description ON DescriptionDetail AFTER INSERT AS
```

```
BEGIN
```

```
    UPDATE Descriptions
```

```
    SET Total = (SELECT SUM(Total) as SUM
```

```
    FROM DescriptionDetail
```

```
    WHERE IdDescription = Descriptions.IdDescription )
```

```
    FROM Descriptions
```

```
    JOIN inserted ON Descriptions.IdDescription = inserted.IdDescription
```

```
END
```

```
GO
```



```
CREATE TRIGGER trg_descriptionupdate ON DescriptionDetail AFTER UPDATE AS  
  
BEGIN  
  
    UPDATE Descriptions  
  
    SET Total = (SELECT SUM(Total) as SUM  
  
    FROM DescriptionDetail  
  
    WHERE IdDescription = Descriptions.IdDescription )  
  
    FROM Descriptions  
  
    JOIN deleted ON Descriptions.IdDescription = deleted.IdDescription  
  
END
```

```
CREATE TRIGGER trg_descriptiondelete ON DescriptionDetail AFTER DELETE AS  
  
BEGIN  
  
    UPDATE Descriptions  
  
    SET Total = (SELECT SUM(Total) as SUM  
  
    FROM DescriptionDetail  
  
    WHERE IdDescription = Descriptions.IdDescription )  
  
    FROM Descriptions  
  
    JOIN deleted ON Descriptions.IdDescription = deleted.IdDescription  
  
    END
```



- Procedure

-- proc add Admin

create proc sp\_insert\_Admin

    @username nvarchar(50),

    @password nvarchar(50)

as

begin tran

    begin try

        insert into dbo.Admin values (@username, @password)

    end try

    begin catch

        ROLLBACK TRANSACTION;

        RETURN;

    end catch

COMMIT TRANSACTION

GO

-- proc update Admin

create proc sp\_update\_Admin

    @newusername nvarchar(50),

    @newpassword nvarchar(50),

    @oldusername nvarchar(50),

    @oldpassword nvarchar(50),

    @idAdmin int

as

begin tran



begin try

```
if exists (select * from dbo.Admin where IdAdmin = @idAdmin and Username =  
@oldusername and Password = @oldpassword )
```

```
if (@newpassword is null)
```

```
update dbo.Admin set Username = @newusername where IdAdmin =  
@idAdmin
```

```
else if (@newusername is null)
```

```
update dbo.Admin set Password = @newpassword where IdAdmin =  
@idAdmin
```

```
else if (@newpassword is not null and @newusername is not null)
```

```
update dbo.Admin set Username = @newusername ,Password =  
@newpassword where IdAdmin = @idAdmin
```

```
else
```

```
raiserror ('Update information is null', 16, 1)
```

end try

begin catch

```
ROLLBACK TRANSACTION;
```

```
RETURN;
```

end catch

```
COMMIT TRANSACTION
```

```
GO
```

```
-- proc update Staff
```

```
create proc sp_update_Staff
```

```
@newusername nvarchar(50),
```

```
@newpassword varchar(50),
```

```
@oldusername nvarchar(50),
```

```
@oldpassword varchar(50),
```



```
@idStaff int

as

begin tran

    begin try

        if exists (select * from dbo.Staff where IdStaff = @idStaff and Username =
@oldusername and Password = @oldpassword )

            if (@newpassword is null)

                update dbo.Staff set Username = @newusername where IdStaff =
@idStaff

            else if (@newusername is null)

                update dbo.Staff set Password = @newpassword where IdStaff =
@idStaff

            else if (@newpassword is not null and @newusername is not null)

                update dbo.Staff set Username = @newusername ,Password =
@newpassword where IdStaff = @idStaff

            else

                raiserror ('Update information is null', 16, 1)

    end try

    begin catch

        ROLLBACK TRANSACTION;

        RETURN;

    end catch

COMMIT TRANSACTION

GO
```





-- proc add Staff

create proc sp\_insert\_Staff

    @username nvarchar(50),

    @password nvarchar(50),

    @fullname nvarchar(50),

    @admin nvarchar(50)

as

begin tran

    begin try

        if exists (select \* from dbo.Admin where dbo.Admin.Username = @admin)

            insert into dbo.Staff values (@admin, @username, @password)

        else

            raiserror ('Failed Auth', 16, 1)

    end try

    begin catch

        ROLLBACK TRANSACTION;

        RETURN;

    end catch

COMMIT TRANSACTION

GO



-- proc add Client

create proc sp\_insert\_Client

    @FullName nvarchar(50),

    @Phone varchar(50),

    @Username varchar(50),

    @hashPassword varchar(50),

    @email varchar(50),

    @number int,

    @street nvarchar(50),

    @ward nvarchar(50),

    @city nvarchar(50),

    @providence nvarchar(50)

as

begin tran

    begin try

        insert into dbo.Client values (@FullName, @Phone, @Username, @hashPassword,  
@Email)

        insert into dbo.UserAddress values (@Username, @number, @street, @ward, @city,  
@providence)

    end try

    begin catch

        ROLLBACK TRANSACTION;

        RETURN;

    end catch

COMMIT TRANSACTION

GO



-- proc update cost

create proc sp\_update\_tour\_cost

@newCostAdults float, @newCostChilds float, @idTour int

AS

begin tran

BEGIN Try

IF not exists (SELECT \* FROM dbo.Tour WHERE IdTour = @idTour)

raiserror ('Không tồn tại tour cần cập nhật', 16, 1)

else

begin

update dbo.Tour

set CostAdo = @newCostAdults, CostChild = @newCostChilds

where IdTour = @idTour

end

END Try

begin catch

ROLLBACK TRANSACTION;

RETURN;

end catch

COMMIT TRANSACTION

GO



Danh sách các TRIGGER và Procedure

	Tên	Chức năng
Trigger	trg_OrderTour	Tính tổng tiền của hóa đơn
	trg_CapNhatOrderTour	Cập nhật tiền hóa đơn
	trg_HuyOrderTour	Hủy Tour
	trg_description	Cập nhật description
	trg_descriptionupdate	Update Description
	trg_descriptiondelete	Delete Description
Tour	sp_insert_Admin	Insert Admin
	sp_update_Admin	Update Admin
	sp_insert_Staff	Insert Staff
	sp_update_Staff	Update Staff
	sp_insert_Client	Insert Client
	sp_update_tour_cost	Update Cost

## Xác định yêu cầu chức năng

### I/ Chức năng:(dự kiến)

1. Trang chủ:
  - Xem danh sách các tour trên trang chủ.
  - Hiện thị tour theo danh sách, tìm kiếm tour theo tên.
  - Đăng nhập + đăng ký.
  - Trang thông tin cá nhân:
  - Xem thông tin cá nhân + địa chỉ (dùng để đón khi đã mua tour) + đơn hàng vừa đặt, thay đổi thông tin cá nhân.
  - Thay đổi địa chỉ đón .
  - Kiểm tra order theo mã order và email
  - Xem danh sách tất cả order đã và đang đặt.
2. Admin:
  - Xem các khoản thu - chi theo tháng, năm.
  - Xem danh sách nhân viên có trong hệ thống (có thể thay đổi chức năng của người dùng ), tìm kiếm người dùng theo tên
  - Xem danh sách khách hàng có trong hệ thống (có thể thay đổi chức năng của người dùng ), tìm kiếm người dùng theo tên
3. Nhân viên:
  - Xem danh sách khách hàng có trong hệ thống (có thể thay đổi chức năng của người dùng ), tìm kiếm người dùng theo tên
  - Thêm tour vào hệ thống
  - Cập nhật thông tin tour.
  - Xem danh sách đơn hàng và tùy chỉnh trạng thái đơn.
  - Xem thu-chi theo ngày/ tháng/ năm, xem doanh thu từ ngày này sang ngày khác.
  - Cập nhật các khoản chi của mỗi tour
4. Giỏ hàng
  - Ước tính giá , thay đổi số lượng (số lượng tour, số lượng người đi) hoặc xóa cả giỏ hàng hoặc xóa 1 tour.
    - Chức năng thanh toán đơn.
  - Chọn hình thức thanh toán, xong tiến hành thanh toán đơn hàng.

## II/ Usecase

	Đối tượng sử dụng - Trang có chức năng	Mô tả
Chức năng tạo đơn hàng mới, . Chỉnh sửa đơn hàng nếu chưa thực hiện thanh toán.	Khách hàng ( Trang chủ - Giỏ hàng)	Khách hàng chọn tour cần đặt, tạo mới đơn, nhập số lượng người đi (bao gồm người lớn và trẻ em), chọn hình thức thanh toán. Nếu như khách hàng chưa đồng ý thanh toán thì có thể thay đổi tour, thay đổi số lượng người đi.
Chức năng đặt hàng chung	Khách hàng (Giỏ hàng)	Khi tiến hành đặt hàng, hệ thống sẽ đưa ra phương thức thanh toán dự theo hình thức mà khách hàng đã chọn, sau đó nhân viên hệ thống sẽ xác nhận lại việc thanh toán và cập nhật trạng thái đơn của khách (đã thanh toán, chưa thanh toán, đang chờ thanh toán)
Chức năng cập nhật, thêm mới các tour	Nhân viên (Trang quản lý - phân hệ nhân viên)	Nhân viên có thể thêm các thông tin tour, cập nhật các tour
Chức năng theo dõi thu - chi, lịch sử các đơn hàng, tình trạng đơn hàng	Admin - Nhân viên (Trang quản lý - phân hệ Admin - Nhân viên)	Phía quản lý có thể xem thống kê doanh thu của tháng này/ năm/ tổng doanh thu., các khoản chi ở từng địa điểm của tour, tổng các khoản chi của một tour, hoặc của nhiều tour theo những mốc thời gian. Tour bán chạy nhất, tour bán ít nhất để dễ dàng tùy chỉnh trong tương lai. Xem được tình trạng của các đơn hàng đang giao
Chức năng theo dõi các thành viên trong hệ thống	Admin	Admin có thể xem được các tài khoản đang có trong hệ thống, có thể là khách hàng, và các admin, nhân viên khác.
Chức năng theo dõi, quản lý đơn hàng	Nhân viên	Nhân viên có thể theo dõi, quản lý các đơn hàng sau khi khách hàng đặt hàng ( nhân viên có nghĩa vụ phải cập nhật trạng thái của đơn hàng và thông báo cho khách hàng nếu cần thiết)
Chức năng theo dõi các khách hàng có trong hệ thống	Nhân viên - Admin	Nhân viên có thể xem được các tài khoản khách hàng đang có trong hệ thống



- Chức năng đăng nhập: phân quyền người dùng( khách hàng, nhân viên, admin).
- Khách hàng khi đăng nhập vào trang thì sẽ chuyển hướng sang trang chủ chứa các thông tin các tour
- Nhân viên khi đăng nhập thành công sẽ chuyển hướng trang quản lý dành cho nhân viên (quản lý tour, doanh thu, khách hàng)
- Admin khi đăng nhập vào trang ví dụ : /wp-admin thì sẽ chuyển hướng vào trang quản lý người dùng, quản lý các nhân viên

## KHÁCH HÀNG

- Chức năng tạo đơn, xác nhận đơn, hoàn tất đơn. Chức năng đánh giá tour, gửi phản hồi sau khi hoàn tất đơn. Chức năng chỉnh sửa đơn hàng nếu đơn hàng chưa được thanh toán. Chức năng gửi email quảng cáo/khuyến mãi (chỉ với những người dùng đăng ký).

## NHÂN VIÊN:

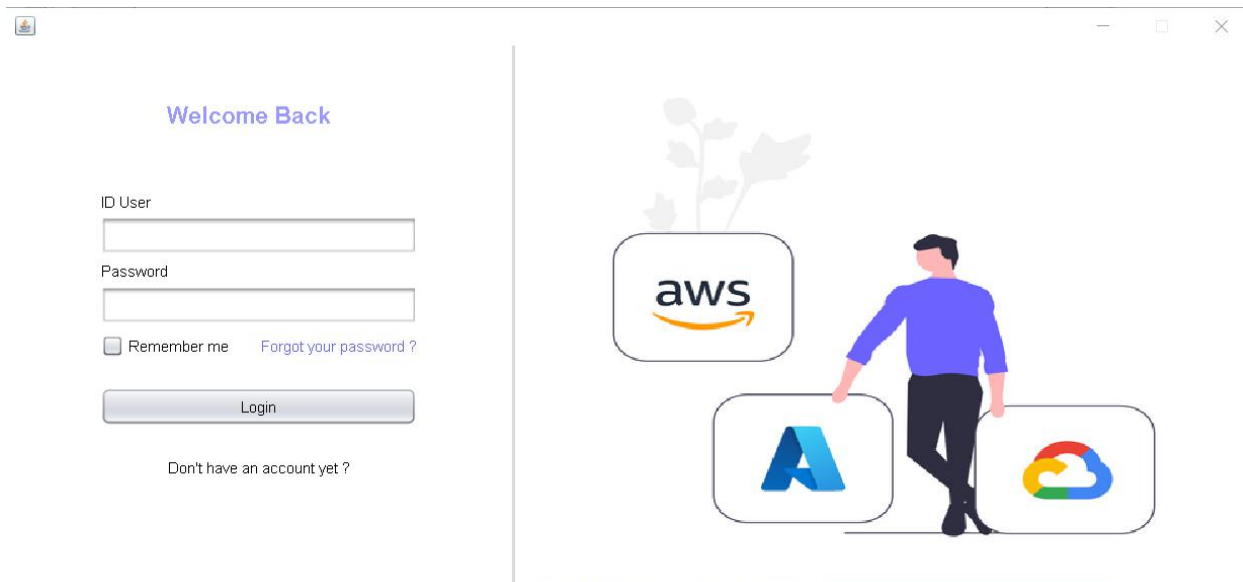
- Nhân viên có thể thực hiện thêm mới, cập nhật, xóa các tour
- Nhân viên có thể thống kê thu - chi/ số lượng đơn hàng.
- Nhân viên có thể quản lý tình trạng các đơn hàng
- Nhân viên có thể khách hàng

## ADMIN ( NHÀ QUẢN LÝ )

- Theo dõi số thành viên trong hệ thống - có thể là các quản lý, các khách hàng, nhân viên (tên tài khoản, email, vai trò, ...)
- Chức năng theo dõi thu-chi
- Theo dõi tình trạng đơn hàng đơn hàng
- Quản lý các tài khoản nhân viên

## Layout của giao diện:

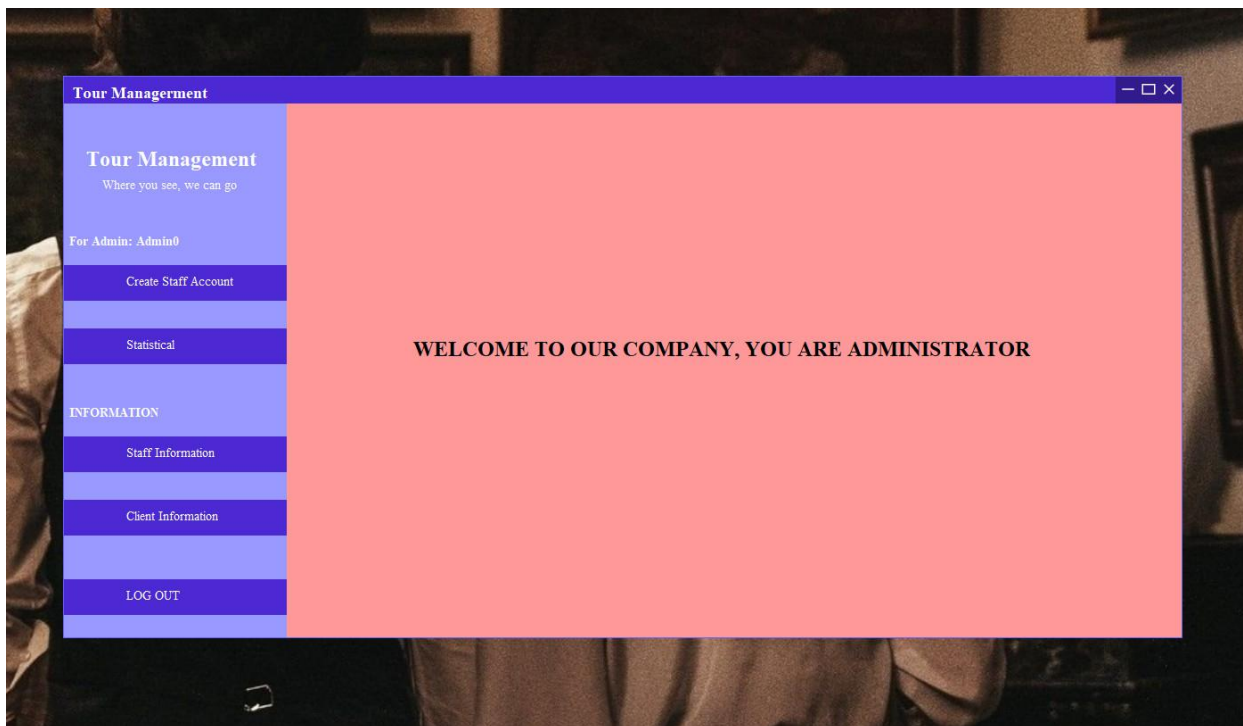
Trang Login



The screenshot shows a login interface with a left sidebar and a main content area. The sidebar contains a 'Welcome Back' message, input fields for 'ID User' and 'Password', a 'Remember me' checkbox, a 'Forgot your password?' link, a 'Login' button, and a link for 'Don't have an account yet?'. The main content area features a large illustration of a person standing next to three boxes containing the AWS, Azure, and Google Cloud logos.

Phân hệ Admin:

Trang chủ:



Trang tạo tài khoản nhân viên





Tour Management

Tour Management

Where you see, we can go

For Admin: Admin0

Create Staff Account

Statistical

INFORMATION

Staff Information

Client Information

LOG OUT

Admin

Admin0

Full Name

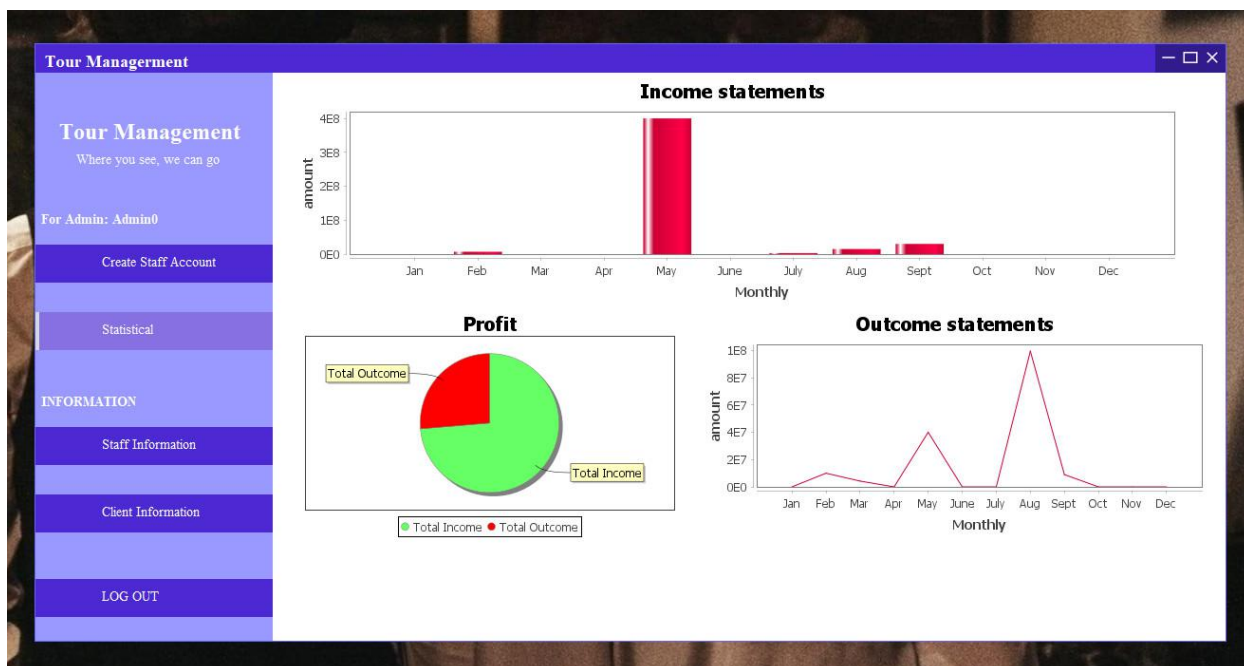
Username

Password

Create Staff

Number of all curent Staff in system now: 200000

Trang Thống kê



Trang Quản lý nhân viên:



**Tour Management**

**Tour Management**  
Where you see, we can go

For Admin: Admin0

Create Staff Account

Statistical

INFORMATION

Staff Information

Client Information

LOG OUT

Staff  Search AC Search DAC

Deactivate Staff: 99996

Id Staff	Full Name	Username	Status
1	Vương Tấn Huy	Staff0	1
2	Liêu Khắc Tuấn	Staff1	1
3	Thăng Quyết Thắng	Staff2	1
4	Mùa Văn Thường	Staff3	1
7	Phù Hà Phương	Staff6	1
8	Lạc Công Lộc	Staff7	1
10	Mac Bích Phương	Staff9	1
11	Hà Khánh Thủy	Staff10	1
12	Hi Bích Thảo	Staff11	1

Active Deactivate

Deactivate Staff: 100004

Id Staff	Full Name	Username	Status
5	Trịnh Đức Huy	Staff4	0
6	Bồ Huy Hoàng	Staff5	0
9	Ninh Thiều Mai	Staff8	0
675	Khiếu Hướng Bình	Staff674	0
100000	Thẩm Hạnh Nhơn	Staff100000	0
100001	Tổng Hướng Thiện	Staff100001	0
100002	Bàng Phú An	Staff100002	0
100003	Hân Duy Uyên	Staff100003	0
100004	Lạc Hương Chi	Staff100004	0

## Trang quản lý khách hàng

**Tour Management**

**Tour Management**  
Where you see, we can go

For Admin: Admin0

Create Staff Account

Statistical

INFORMATION

Staff Information

Client Information

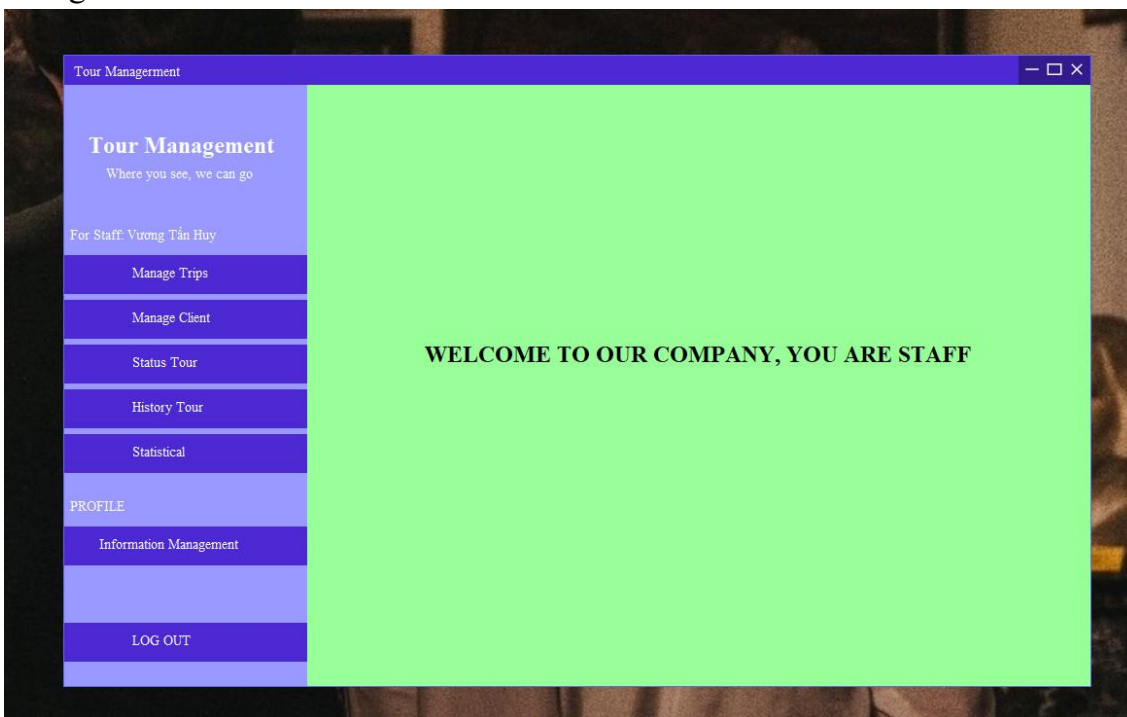
LOG OUT

Client  Search

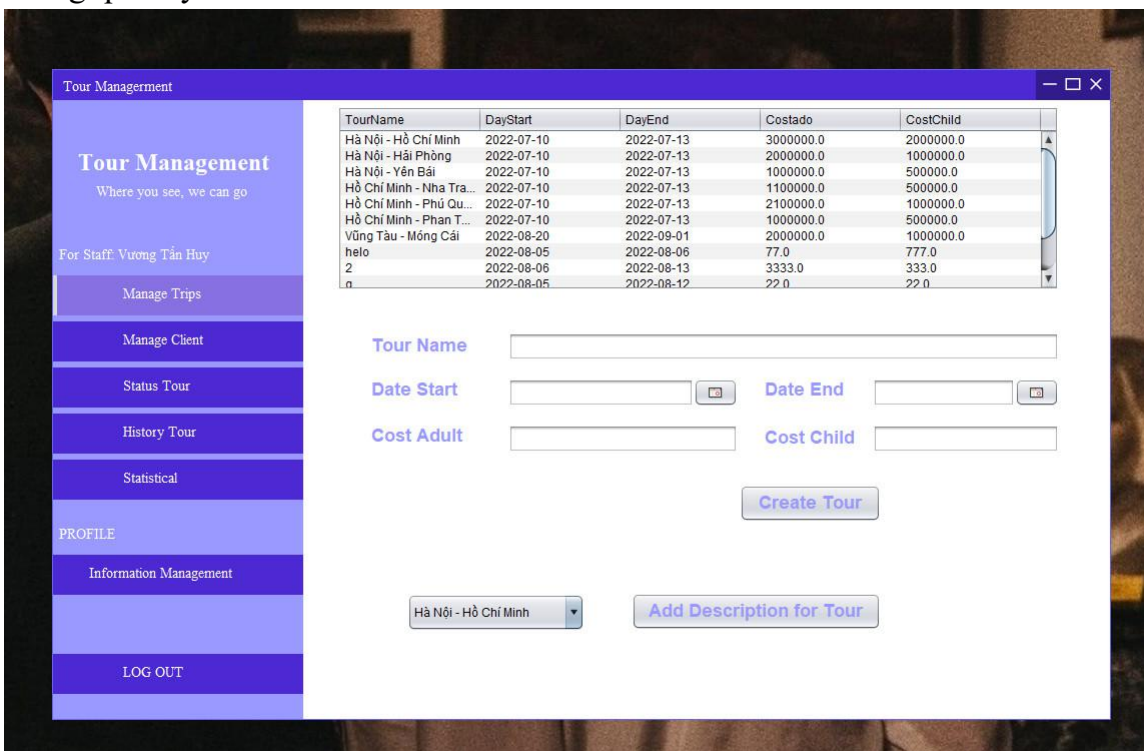
Name	Address	Phone	Username	Phone
Mã Ng?c C?nh	2, Lê Lợi, Phu?ng 1, ...	274441626145	Client0	client0@gmail.com
Từ Cường Thịnh	2, Lê Duẩn, Phu?ng ...	9684953416	Client1	client1@gmail.com
Đái Giang Nam	2, Hai Bà Trưng, Phu...	3740693234	Client2	client2@gmail.com
Anh Quang	null, null, null, null, null	0926287415	Client3	client3@gmail.com
Trương Huệ An	null, null, null, null, null	3520452055	Client4	client4@gmail.com
Cổng Kiệt Hồng	null, null, null, null, null	3156120058	Client5	client5@gmail.com
Hứa Anh Quân	null, null, null, null, null	9718760251	Client6	client6@gmail.com
Đình Diễm Phương	null, null, null, null, null	3912570724	Client7	client7@gmail.com
Cổ Minh Duyên	null, null, null, null, null	3914840287	Client8	client8@gmail.com
Ấu Hữu Nam	null, null, null, null, null	5957703031	Client9	client9@gmail.com
Ngô Thanh Loan	null, null, null, null, null	3011392952	Client10	client10@gmail.com
Chung Ánh Hồng	null, null, null, null, null	5663166720	Client11	client11@gmail.com
Ung Trúc Vân	null, null, null, null, null	8118763314	Client12	client12@gmail.com
Đường Ngọc Huy	null, null, null, null, null	4816906554	Client13	client13@gmail.com
Võ Thiện Luân	null, null, null, null, null	5267236567	Client14	client14@gmail.com
Chung Duy Cẩn	null, null, null, null, null	8457463313	Client15	client15@gmail.com
Cổ Thủy Hạnh	null, null, null, null, null	3886168880	Client16	client16@gmail.com
Cổ Quốc Hạnh	null, null, null, null, null	8593930590	Client17	client17@gmail.com
Quản Trung Dũng	null, null, null, null, null	5114762586	Client18	client18@gmail.com
Hầu Khánh Hải	null, null, null, null, null	2507743372	Client19	client19@gmail.com
Giao Xuân Sơn	null, null, null, null, null	9490285414	Client20	client20@gmail.com
Ân Chấn Hùng	null, null, null, null, null	7477422674	Client21	client21@gmail.com
Khu Huệ Linh	null, null, null, null, null	1546332715	Client22	client22@gmail.com

## Phân hệ Nhân viên:

Trang chủ:



Trang quản lý tour:





## Trang quản lý khách hàng:

The screenshot shows a web application window titled "Tour Management". On the left is a sidebar menu with options: "Manage Trips", "Manage Client" (selected), "Status Tour", "History Tour", "Statistical", "PROFILE", "Information Management", and "LOG OUT". The main content area is titled "Manage Client" and contains a form with the following fields:

- FullName: Công Kiệt Hồng
- Phone: 3156120058
- UserName: Client5
- Password: pass5
- Email: client5@gmail.com

Below the form are four buttons: "Insert", "Update", "Delete", and "Clear". At the bottom, there is a table listing clients:

Id	Fullname	Phone	Username	Password	Email
4	Anh Quang	0926287415	Client3	pass3	client3@gmail.c...
5	Trương Huệ An	3520452055	Client4	pass4	client4@gmail.c...
6	Công Kiệt Hồng	3156120058	Client5	pass5	client5@gmail.c...
7	Hứa Anh Quân	9718760251	Client6	pass6	client6@gmail.c...
8	Đinh Diễm Phư...	3912570724	Client7	pass7	client7@gmail.c...
9	Cổ Minh Duyên	3914840287	Client8	pass8	client8@gmail.c...
10	Ấu Hữu Nam	5957703031	Client9	pass9	client9@gmail.c...

## Trang quản lý trạng thái tour:



Tour Management

Tour Management

Where you see, we can go

For Staff: Vương Tấn Huy

Manage Trips

Manage Client

Status Tour

History Tour

Statistical

PROFILE

Information Management

LOG OUT

Status Tour Management

Search

Attention: For change status tour, you must find by customer name !!!

Customer Username	Customer Fullname	Order ID	Status
Mã Ng?c C?nh	Client0	5	Đã thanh toán
Mã Ng?c C?nh	Client0	6	Chờ thanh toán
Mã Ng?c C?nh	Client0	7	Đã thanh toán
Mã Ng?c C?nh	Client0	9	Chưa thanh toán

Wait for payment

Already Paid

Trang quản lý lịch sử giao dịch:

Tour Management

Tour Management

Where you see, we can go

For Staff: Vương Tấn Huy

Manage Trips

Manage Client

Status Tour

History Tour

Statistical

PROFILE

Information Management

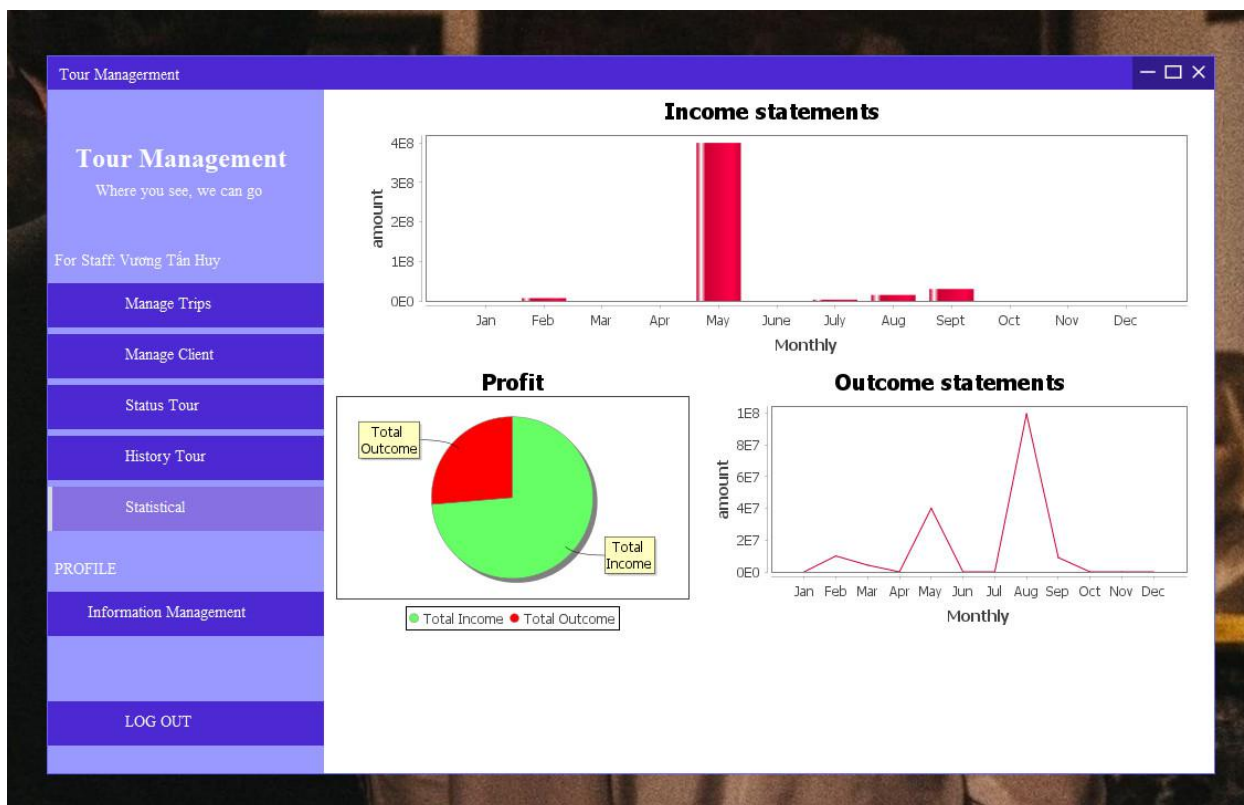
LOG OUT

History Payment

Search

Customer Fullname	Customer Username	Order ID	Payment day
Mã Ng?c C?nh	Client0	5	2022-08-05 21:04:45.883
Mã Ng?c C?nh	Client0	7	2022-08-05 22:40:52.33

Trang thống kê:



Trang quản lý thông tin cá nhân

The screenshot displays the 'Profile Management' page within the 'Tour Management' interface. It includes a sidebar menu and a main content area with the following elements:

- Profile Management:** A section showing current user information:
  - Username:** Staff0
  - Fullname:** Vương Tấn Huy
  - Your Admin:** Admin5863
- Update Profile:** A section with three input fields and corresponding 'Update' buttons:
  - Username:** [Input Field] [Update]
  - Fulname:** [Input Field] [Update]
  - Password:** [Input Field] [Update]



## Phân hệ Khách hàng: Trang chủ

Tour Management

Tour Management

Where you see, we can go

For Client: Mã Ngọc Cảnh

Book Trip Now

Your Trip

Information

LOG OUT

WELCOME, LETS CHOOSE YOUR TRIPS

## Trang chọn Tour

Tour Management

Tour Management

Where you see, we can go

For Client: Mã Ngọc Cảnh

Book Trip Now

Your Trip

Information

LOG OUT

IdTour	Tour Name	DayStart	DayEnd	Address Start	Vehicle
1	Hà Nội - Hồ Chí Minh	2022-07-10	2022-07-13	Long Biên	hhh
1	Hà Nội - Hồ Chí Minh	2022-07-10	2022-07-13	Long Biên	hhh
1	Hà Nội - Hồ Chí Minh	2022-07-10	2022-07-13	Cầu Giấy	ggg
1	Hà Nội - Hồ Chí Minh	2022-07-10	2022-07-13	Cầu Giấy	ggg
2	Hà Nội - Hải Phòng	2022-07-10	2022-07-13	Long Biên	hhh
2	Hà Nội - Hải Phòng	2022-07-10	2022-07-13	Long Biên	hhh
2	Hà Nội - Hải Phòng	2022-07-10	2022-07-13	Thanh Xuân	Tau Hoa
2	Hà Nội - Hải Phòng	2022-07-10	2022-07-13	Thanh Xuân	Tau Hoa
3	Hà Nội - Yên Bái	2022-07-10	2022-07-13	Long Biên	hhh
3	Hà Nội - Yên Bái	2022-07-10	2022-07-13	Long Biên	hhh
4	Hồ Chí Minh - Nha Trang	2022-07-10	2022-07-13	Quận 1	
4	Hồ Chí Minh - Nha Trang	2022-07-10	2022-07-13	Quận 1	

Chose for Detail 

helo

Detail Description

alo

Phố cổ Hà Nội

Tắm biển Đồ Sơn

Đảo Cát Bà

Lẩu dê Thái Bình

Chose Tour you want to Order 

Hà Nội - Hồ Chí Minh

Number of Adult

Number of Child

Register Tour

## Trang đặt tour

Tour Management

Where you see, we can go

For Client: Mã Ng?c C?nh

Book Trip Now

Your Trip

Information

LOG OUT

Tour	Number of adults	Number of childs	Total
hh	8	9	400000000
Hà Nội - Hồ Chí Minh	3	3	15000000

hh

Number of Adult

Number of Child

Update

Delete

Tour Management

Where you see, we can go

For Client: Công Kiệt Hồng

Book Trip Now

Your Trip

Information

LOG OUT

Name: Công Kiệt Hồng

Address: null, null, null, null, null

Phone: 3156120058

Username: Client5

Gmail: client5@gmail.com

Name	Address	Phone	Username	Phone
Công Kiệt Hồng	ll, null, null, null, null	3156120058	Client5	client5@gmail.com

Update



# Partiton

## Phân chia dữ liệu – Partition

### I. Partition:

- *Partition (Phân chia dữ liệu)*: là một giải pháp chia nhỏ một Database lớn thành nhiều Database nhỏ, ta có thể phân tách từng bảng hoặc cả một DB ra nhiều phần nhỏ đặt ở nhiều máy chủ (server) khác nhau. Điều này sẽ giúp cho hệ thống DB của chúng ta đạt được các tính chất khả năng bảo trì, hiệu xuất, tính sẵn sàng, và cân bằng tải của ứng dụng. Và giải pháp này cũng giảm chi phí cũng như tính mở rộng để scale up DB bằng cách dùng nhiều server nhỏ gộp lại hơn là nâng cấp một server lớn.
- *Partition function*: *Ánh xạ các dòng của bảng hoặc các chỉ mục trong phân mảnh dựa theo tiêu chí phân mảnh*. Qui định giá trị biên cho các đoạn. Hệ thống dựa vào hàm này để xác định đoạn mà mỗi bản ghi thuộc vào.
- *Partition Scheme*: *Ánh xạ các phân mảnh của bảng vào các filegroup (mỗi filegroup chứa một phân mảnh)*

### II. Lợi ích của việc sử dụng Partition:

- Cải thiện khả năng co giãn và khả năng quản lý của các bảng lớn (large table) và các bảng có các kiểu truy cập khác nhau.
- Khi các bảng và các chỉ mục quá lớn, việc phân mảnh giúp chia dữ liệu thành các phần nhỏ hơn, có thể quản lý được.
- Nếu 1 bảng lớn tồn tại trên hệ thống nhiều CPUs, việc phân mảnh bảng sẽ giúp tăng hiệu suất khi thực hiện song song
- *Việc backup/restore* một đoạn mà không ảnh hưởng đến các đoạn còn lại.

➔ Phân chia dữ liệu sẽ giúp cho hệ thống DB đạt được các tính chất khả năng bảo trì, hiệu xuất, tính sẵn sàng, và cân bằng tải của ứng dụng. Và giải pháp này cũng giảm chi phí cũng như tính mở rộng để scale up DB bằng cách dùng nhiều server nhỏ gộp lại hơn là nâng cấp một server lớn.

### III. Các kiểu phân chia dữ liệu:

- Phân chia theo chiều ngang (horizontal partition):
  - Phân chia các dòng trong một table thành nhiều table khác nhau.
  - Tình huống áp dụng: khi nhiều người dùng khác nhau cần truy cập các dòng dữ liệu khác nhau.
  - Ưu điểm: tối ưu hoá tốc độ truy cập dữ liệu
  - Nhược điểm:

Phức tạp khi phải truy cập toàn bộ dữ liệu.

Nếu không phân chia cẩn thận sẽ dễ gây mất cân bằng giữa các DB.

- Phân chia theo chiều dọc (vertical partition):
  - Phân chia một cấu trúc luận lý thành những cấu trúc lưu trữ vật lý khác nhau. Phân chia dựa vào thuộc tính (cột) của bảng thành nhiều bảng khác nhau.
  - Tình huống áp dụng: chia bảng thành các loại cấu trúc có giá trị sử dụng khác nhau hoặc cấu trúc truy cập thường xuyên và không thường xuyên.



- Ưu điểm: Dễ thực hiện và không ảnh hưởng đến ứng dụng
- Nhược điểm: Khi hệ thống lớn thì dữ liệu cũng lớn theo, do đó phải thực hiện phân chia nhiều lần.

#### IV. Phân chia dữ liệu bảng Client(Hoá đơn) theo chiều ngang

##### **Bước 1:** Tạo các file group

- Tạo các filegroup để chứa các phân mảnh sau khi đã phân chia.
- Group 1 (Primary): lưu các OrderTour được tạo trước năm 2021
- Group 2 (OrderTour\_2021): lưu các OrderTour được tạo trong khoảng 2021 - 2022 (Order trong năm 2021)
- Group 3 (OrderTour\_2022): lưu các OrderTour được tạo sau năm 2021

```
ALTER DATABASE ManageTourist ADD FILEGROUP OrderTour_2021
```

```
ALTER DATABASE ManageTourist ADD FILEGROUP OrderTour_2022
```

```
ALTER DATABASE ManageTourist ADD FILE ( NAME = N'OrderTour_2021',
```

```
FILENAME =
```

```
N'F:\Partition\OrderTour_Year\OrderTour_2021.ndf',
```

```
SIZE = 10MB,
```

```
MAXSIZE = UNLIMITED,
```

```
FILEGROWTH = 1024 KB)
```

```
TO FILEGROUP OrderTour_2021
```

```
ALTER DATABASE ManageTourist ADD FILE ( NAME = N'HoaDon_2022',
```

```
FILENAME =
```

```
N'F:\Partition\OrderTour_Year\OrderTour_2022.ndf',
```

```
SIZE = 10MB,
```

```
MAXSIZE = UNLIMITED,
```

```
FILEGROWTH = 1024 KB)
```

```
TO FILEGROUP OrderTour_2022
```

##### **Bước 2:** Tạo Partition Function

- Tạo các giá trị biên cho mỗi bảng dựa vào cột createDate trong bảng chính (OrderTour)



- Hàm Partition function ‘Order\_by\_Year\_function’ tạo được 3 bảng nhỏ chia dữ liệu theo các mốc thời gian: trước ngày 1/1/2021; từ 1/1/2021 đến 31/12/2021; từ 1/1/2022 trở về sau.
- RANGE RIGHT nghĩa là lấy < or >=  
LEFT lấy <= and >

<1/1/2021	PRIMARY
>=1/1/2021 and <31/12/2021	OrderTour_2021
>=1/1/2022	OrderTour_2022

```
CREATE PARTITION FUNCTION OrderTour_by_Year_Function (datetime)
AS
RANGE LEFT
FOR VALUES ('2020-12-31','2021-12-31')
GO
```

### **Bước 3: Tạo partition schema**

- Hàm Partition Scheme ‘OrderTour\_by\_Year\_Scheme’ dùng hàm ‘OrderTour\_by\_Year\_Function’ để “gắn” các đoạn vào từng filegroup

```
CREATE PARTITION SCHEME OrderTour_by_Year_Scheme
AS
PARTITION OrderTour_by_Year_function
TO ([PRIMARY], OrderTour_2021, OrderTour_2022 )
GO
```



#### **Bước 4: Tạo partitioned table/index**

```
CREATE TABLE OrderTour (  
    IdOrder INT IDENTITY(1,1) PRIMARY KEY NONCLUSTERED ON [PRIMARY],  
    IdClient int,  
    createDate datetime default getdate(),  
    Total float,  
    isDeleted int default 0)  
  
GO  
  
CREATE CLUSTERED INDEX ix_OrderTour_by_Year  
ON dbo.OrderTour  
(  
    [createDate]  
) ON [OrderTour_by_Year_scheme]([createDate])
```



## Cài đặt ứng dụng

**Link GitHub cho SourceCode:**

**<https://github.com/kubi2811/Tourism-Management.git>**