📌 Đề tài: Travel/Tour Booking System

# 1. 🎯 Mục tiêu

Xây dựng hệ thống quản lý đặt tour du lịch cho khách hàng, cho phép đặt tour cá nhân/nhóm, quản lý hướng dẫn viên, thanh toán. Đồng thời, hệ thống cung cấp báo cáo thống kê về tour và khách hàng.

# 2. ⚙️ Cách thức vận hành

## 2.1. Khách hàng

* Đăng nhập thông tin
* Đăng ký Tour (đăng ký theo cá nhân hoặc theo nhóm)
* Xác nhận thanh toán theo số người đã đăng ký

## 2.2. Quản trị viên (phải có quyền truy cập của quản trị viên)

* Quản lý tour
* Quản lý thông tin tour
* Nhập thông tin Tour vào hệ thống

## 2.3. Hướng dẫn viên (phải có quyền truy cập của hướng dẫn viên)

* Chọn Tour dẫn (một hướng dẫn viên có thể chọn nhiều tour dẫn miễn là không trùng nhau)
* Nếu tour đạt yêu cầu về số lượng người tham gia cũng như hướng dẫn viên dẫn tour sẽ được xác nhận đưa vào thực thi. Các hướng dẫn viên sẽ nhận được danh sách khách hàng đăng ký Tour mình dẫn

## 2.4. Chu trình hoạt động của chương trình

* Chương trình nhận thông tin Tour của quản trị viên nhập vào qua giao diện Gui và lưu các thông tin Tour vào Database
* Khách đăng ký thông tin và đăng ký Tour dựa vào các Tour đã được quản trị viên đưa vào trước đó và đăng ký số người.
* Khách hàng thanh toán sẽ được đưa vào danh sách đăng ký Tour và được lưu vào dưới dạng thông tin đăng ký BookingTour
* Hướng dẫn viên cũng đăng ký thông tin cá nhân và chọn Tour, một hướng dẫn viên có thể chọn nhiều Tour dẫn miễn là không trùng nhau
* Khi Tour đủ 80% số khách hàng yêu cầu và có đủ số hướng dẫn viên thì Tour có thể thực thi và sẽ được tạo ra danh sách gồm các khách hàng và hướng dẫn viên đăng ký Tour đó và được lưu vào database

**\* Note: các thông tin liên quan đến tour sẽ được trao đổi với khách hàng qua email và số điện thoại**

# 3. 🧱 Các đối tượng chính (OOP Classes)

## Human

* Attributes:
* Id: String
* Name: String
* Birthday: String
* phoneNumber: String
* Email: String
* Methods:
* Human()
* chuanHoaTenVaNgaySinh(): void

## Tour

* Attributes: tourId tạo tự động: TOUR001
* tourId: String
* tourName: String
* destination: String
* dayStart: String
* numberOfDays: int
* price : float
* numberOfPassengers: int
* tourState: String
* numberOfGuides: int
* tourGuide: String[ ] arr
* Methods:
* Tour ()
* getTourId(): String
* getTourName(): String
* getFrom(): String
* getDestination(): String
* getDayStart(): String
* getNumberOfDays(): float
* getPrice(): float
* getNumberOfPassengers(): int
* getTourState(): String
* getTourGuide(): String (tên các hướng dẫn viên / chưa có hướng dẫn viên)
* toString(): String
* checkCustomerAvailability(): Boolean
* checkGuideAvailability(): Boolean

## Customer (kế thừa các thuộc tính từ Human)

* Attributes: Id tạo tự động: KH001
* Id: String
* Birthday: String
* Name: String
* phoneNumber: String
* Email: String
* Methods:
* Customer()
* getId(): String
* getName(): String
* getBirthday(): String
* getPhoneNumber(): String
* getEmail(): String

## Booking

* Attributes: bookingId tạo tự động: BK001
* bookingId: String
* tourId: String
* customerId: String
* bookingDate: String
* bookingState: String
* Methods:
* Booking()
* getBookingId(): String
* getTourId(): String
* getCustomerId(): String
* getBookingDate(): String
* getBookingState(): String

## Guide (Kế thừa các thuộc tính từ Human)

* Attributes: Id tạo tự động: HDV001
* Id: String
* Name: String, Birthday: String
* phoneNumber: String
* Email: String
* guideExperience: float
* foriegnLanguage: String[ ] arr
* tours: String[ ] arr
* Methods:
* getld(): String
* getName(): String
* getGuideExperience(): float
* getForiegnLanguage: String[ ] arr
* getToursList: String[] arr

# 4. 🗄️ Database Design

## Tour

* tourId: varchar (primary key)
* tourName: varchar
* destination: varchar
* dayStart: date
* numberOfDays: float
* price: decimal
* numberOfPassenger: int
* tourState: enum(“Đã đầy”, “Còn suất”)
* numberOfGuides: int
* tourGuide: varchar

## Customer

* Id: varchar (primary key)
* Name: varchar
* Birthday: varchar
* phoneNumber: varchar
* Email: varchar

## Booking

* bookingId: varchar (primary key)
* tourId: varchar (match với tour\_id trong table Tour)
* customerId: varchar (match với customer\_id trong table customer)
* bookingDate: date
* bookingState: enum(“Chưa thanh toán”, “Đã thanh toán”)

## Guide

* Id: varchar [primary key]
* Name: varchar
* Birthday: date
* phoneNumber: varchar
* Email: varchar
* guideExperience: float
* foreignLanguage: varchar
* toursList: varchar

## TourGuide

* tourId varchar
* guideId varchar

# 4. ⚙️ Cách thức vận hành

## 4.1. Khách hàng

* Đăng nhập thông tin
* Đăng ký Tour (đăng ký theo cá nhân hoặc theo nhóm)
* Xác nhận thanh toán theo số người đã đăng ký

## 4.2. Quản trị viên (phải có quyền truy cập của quản trị viên)

* Quản lý tour
* Quản lý thông tin tour
* Nhập thông tin Tour vào hệ thống

## 4.3. Hướng dẫn viên (phải có quyền truy cập của hướng dẫn viên)

* Chọn Tour dẫn (một hướng dẫn viên có thể chọn nhiều tour dẫn miễn là không trùng nhau)
* Nếu tour đạt yêu cầu về số lượng người tham gia cũng như hướng dẫn viên dẫn tour sẽ được xác nhận đưa vào thực thi. Các hướng dẫn viên sẽ nhận được danh sách khách hàng đăng ký Tour mình dẫn

## 4.4. Chu trình hoạt động của chương trình

* Chương trình nhận thông tin Tour của quản trị viên nhập vào qua giao diện Gui và lưu các thông tin Tour vào Database
* Khách đăng ký thông tin và đăng ký Tour dựa vào các Tour đã được quản trị viên đưa vào trước đó và đăng ký số người.
* Khách hàng thanh toán sẽ được đưa vào danh sách đăng ký Tour và được lưu vào dưới dạng thông tin đăng ký BookingTour
* Hướng dẫn viên cũng đăng ký thông tin cá nhân và chọn Tour, một hướng dẫn viên có thể chọn nhiều Tour dẫn miễn là không trùng nhau
* Khi Tour đủ 80% số khách hàng yêu cầu và có đủ số hướng dẫn viên thì Tour có thể thực thi và sẽ được tạo ra danh sách gồm các khách hàng và hướng dẫn viên đăng ký Tour đó và được lưu vào database

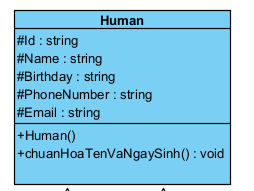
**\* Note: các thông tin liên quan đến tour sẽ được trao đổi với khách hàng qua email hoặc số điện thoại**

# 5. 🔧 Công nghệ triển khai

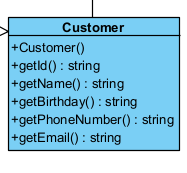
* Ngôn ngữ: Java
* Database: MySQL
* Giao diện:
* JavaSwing

# 6. Class Diagram

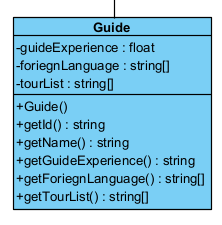
**Class Human:**



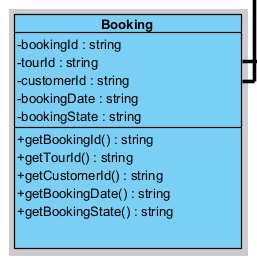
**Class Customer:**



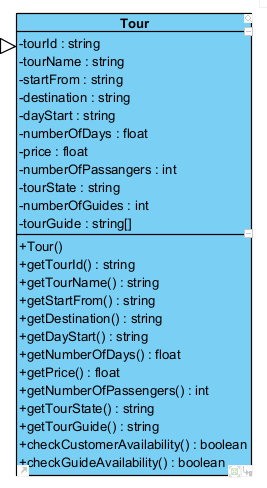
**Class Guide:**



**Class Booking:**



**Class Tour:**

****